
HappyMovie: Aplicación Facebook para recomendar películas para grupos



PROYECTO FIN DE CARRERA

Javier Bécares García, Alejandro Gómez Galindo y Enrique Ortiz
Sánchez

Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid

Dirigido por:
Belén Díaz Agudo
Codirigido por:
Juan A. Recio García

Curso 2010-2011

HappyMovie: Aplicación Facebook para recomendar películas para grupos

Ingeniería del Software e Inteligencia Artificial

Versión 1.0+

Departamento de Ingeniería del Software e Inteligencia
Artificial

Facultad de Informática

Universidad Complutense de Madrid

Dirigido por:

Belén Díaz Agudo

Codirigido por:

Juan A. Recio García

Curso 2010-2011

Copyright © Bécares García, Javier, Gómez Galindo, Alejandro, Ortiz
Sánchez, Enrique

Agradecimientos

Queremos agradecer a nuestros coordinadores de proyecto Belén Díaz Agudo y Juan Antonio Recio García, y a Lara Quijano Sánchez, dado que este proyecto forma parte de su tesis y entre todos hemos dado forma a este trabajo de forma satisfactoria. También queremos dar las gracias a la Facultad de Informática por darnos soporte para las partes técnicas del proyecto.

También queremos dar las gracias a todos los compañeros y amigos que nos han ayudado a la hora de contestar test y usar la aplicación para comprobar su funcionamiento tanto en el correcto almacenamiento de datos como de recomendaciones.

Gracias a todos por vuestra ayuda.

Resumen

El objetivo de este proyecto ha sido la especificación, diseño e implementación de un sistema recomendador grupal para ir al cine, integrado en la red social *Facebook* a través de una aplicación llamada *HappyMovie*. Se han usado técnicas de razonamiento basado en casos, en inglés “Case Based Reasoning”. Nuestro sistema *HappyMovie* trata de paliar ciertas limitaciones existentes en los sistemas de recomendaciones grupales, como son la obtención del perfil de los usuarios (requiere el esfuerzo y el tiempo de los usuarios) y ofrecer recomendaciones para que los usuarios lleguen a un acuerdo final con mayor facilidad. El método que se utilizará para realizar la recomendación al grupo será adaptativo, eligiéndose automáticamente en función de las características del grupo, tamaño, homogeneidad de personalidades, personalidad predominante... Por adaptativo entendemos que la técnica de recomendación variará entre dos técnicas propuestas en el sistema:

- Basada en delegación
- Basada en influencia

y la función de agregación seleccionada. Además hemos mejorado la experiencia del usuario usando tecnologías RIA (*Rich Internet Applications*) para facilitar la interacción con el sistema.

Palabras clave : Sistemas de recomendación, grupos, Redes sociales, personalidad, confianza

The goal of this project was the specification, design and implementation of a group recommender system to go to the cinema, integrated into the social network *Facebook* through an application called *HappyMovie*. Reasoning techniques based on cases, “Case Based Reasoning”, have been used. This group recommender system for films tries to mitigate certain limitations in the group recommender systems, such as obtaining the user profile (it requires the effort and the time of the users) and offering trading methods for users in order to reach the final agreement. The method used to make the recommendation to the group will be adaptive and it will be chosen automatically according to the characteristics of the group, its size, the homogeneity of the personalities, dominant personality in the group), and so on. By adaptive we mean that the recommendation technique varies from:

- Delegation based
- Influence based

and to the selected aggregation function. Moreover we improve the user experience by using Rich Internet Applications technologies to facilitate the interaction with the system.

Keywords: Recommender systems, groups, Social Networks, Personality, Trust

Índice

Agradecimientos	v
Resumen	vii
1. Motivación y objetivos	1
1.1. Descripción	1
1.2. Objetivos	3
1.3. Estructura de la memoria	5
2. Estado del Arte	7
2.1. Sistemas Recomendadores	7
2.1.1. Recomendadores individuales	8
2.1.2. Recomendadores grupales	15
2.2. Redes sociales	22
2.2.1. Facebook	29
2.3. Tecnologías usadas	39
2.3.1. PHP	39
2.3.2. JavaScript	40
2.3.3. AJAX	41
2.3.4. JavaBridge	42
2.3.5. SQL	43
3. El prototipo	47
3.1. Teoría. Modelo teórico	47
3.2. Desarrollo del prototipo	48
3.2.1. Nuestra aplicación: <i>HappyMovie</i>	48
3.2.2. Fases de desarrollo de la aplicación <i>HappyMovie</i>	51
3.2.3. Arquitectura. Módulos <i>HappyMovie</i>	61
3.3. Descripción funcional del sistema. Casos de uso	86
3.4. Funcionamiento global de <i>HappyMovie</i>	86
4. Conclusiones. Líneas de trabajo futuro	103

4.1. Conclusiones y contribución	103
4.2. Líneas de trabajo futuro	104
Bibliografía	109

Índice de figuras

2.1. Esquema de funcionamiento básico de un Sistema Recomendador	8
2.2. Modelo de un sistema de filtrado colaborativo	10
2.3. Fases del ciclo CBR	12
2.4. Descripción de la arquitectura Fab	14
2.5. Mapa de una comunidad virtual. Distinguimos el núcleo de operación formado por usuarios líderes y regulares de la periferia integrada por el resto	23
2.6. Imagen del juego FarmVille	34
2.7. Aplicaciones de test de personalidad	35
2.8. <i>Events</i>	36
2.9. <i>create event</i>	37
2.10. <i>Guest list event</i>	38
2.11. <i>Guest list</i>	38
2.12. <i>Event</i>	39
2.13. Funcionamiento de Javabridge en <i>HappyMovie</i>	43
2.14. MySQL Front	45
2.15. Heidi SQL	45
3.1. Amigos conectados a través de Facebook	49
3.2. Formulario crear Evento	50
3.3. Evento con recomendación	52
3.4. Aplicación <i>Developer</i>	53
3.5. Nueva aplicación	54
3.6. Integración en Facebook	54
3.7. Opciones avanzadas	55
3.8. Organigrama <i>HappyMovie</i>	56
3.9. Máquina virtual	58
3.10. Bases de datos <i>Usuarios</i>	60
3.11. Bases de datos <i>Evento</i>	60
3.12. Arquitectura <i>HappyMovie</i>	61

3.13. Página test de personalidad	66
3.14. Ejemplo de preguntas realizadas en el test TKI	67
3.15. Factores Trust	69
3.16. <i>Interfaz test de preferencias</i>	74
3.17. <i>Interfaz test de preferencias</i>	75
3.18. Archivo test de preferencias	77
3.19. Archivo .xml con las películas de la cartelera	78
3.20. Archivo resultado del recomendador individual	78
3.21. Recomendaciones individuales de todos los asistentes al evento	79
3.22. Diagrama de tablas de Base de Datos de <i>HappyMovie</i>	87
3.23. Diagrama de tablas de Web Crawler	87
3.24. Página inicial <i>HappyMovie</i>	96
3.25. Página inicial <i>HappyMovie</i>	97
3.26. Página de invitaciones.	98
3.27. Evento en <i>HappyMovie</i>	100
3.28. Invitar amigos en <i>HappyMovie</i>	101
3.29. Borrarse de un evento en <i>HappyMovie</i>	102

Índice de Tablas

3.1. Coeficientes usados para calcular el CMW.	62
3.2. Caso de Uso 1. Acceder a la aplicación.	88
3.3. Caso de Uso 2. Abrir la aplicación	88
3.4. Caso de Uso 3. Crear un nuevo evento.	89
3.5. Caso de Uso 4. Invitar amigos al evento.	90
3.6. Caso de Uso 5. Mostrar solicitud de unirse a un evento.	90
3.7. Caso de Uso 6. Aceptar unirse a un evento.	91
3.8. Caso de Uso 7. Acceder a un evento en el que el usuario ya está incluido	92
3.9. Caso de Uso 8. Borrarse de un evento.	93
3.10. Caso de Uso 9. Rellenar test de personalidad.	93
3.11. Caso de Uso 10. Rellenar test de preferencias.	94
3.12. Caso de Uso 11. Finalización del deadline para unirse al evento.	94
3.13. Caso de Uso 12. Votación de películas propuestas	95
3.14. Caso de Uso 13. Finalización del evento.	95

Capítulo 1

Motivación y objetivos

1.1. Descripción

En la actualidad es difícil encontrar un sitio web, principalmente de carácter comercial, que no provea un sistema recomendador que guíe al usuario en la adquisición de uno o varios productos de interés. El crecimiento vertiginoso en volumen y complejidad de la información, y dada la amplia gama de actividades de búsqueda y comercio electrónico que puede realizar diariamente una persona ha propiciado la aparición de estos sistemas. Acertamos al decir que los sistemas de recomendación se han convertido en todo un fenómeno, y están marcando un punto de inflexión ente la segunda generación de servicios de Internet (Web 2.0), vigente desde el año 2004 aproximadamente, y la que será la tercera (Web 3.0, o también denominada “Web Inteligente”).

La Web Inteligente engloba diversos servicios: la web semántica, la minería de datos, las búsquedas utilizando lenguaje natural, las tecnologías de inteligencia artificial, y también, como acabamos de comentar, los sistemas recomendadores. Estos últimos forman parte de una técnica de filtrado de información mediante la cual comparan una colección de datos de interés de un usuario, es decir, sus gustos, preferencias u objetivos, con una colección similar, creando un listado de productos o servicios que a continuación serán recomendados al usuario. Las técnicas usadas están inspiradas en otras pertenecientes a áreas del conocimiento como la Recuperación de Información y el Aprendizaje Automático, haciendo uso de modelos estadísticos o de heurísticas que intentan capturar las correlaciones entre usuarios y objetos.

Visto el éxito que cosecharon los sistemas que podían recomendar al usuario un determinado producto, el paso lógico siguiente fue conseguir recomendar a un grupo de usuarios. Muchas de las actividades que realiza un individuo lo hace en compañía de otras personas. La recomendación para grupos no se puede realizar como se venía haciendo con la recomendación

individual, ya que ahora no sólo se deben tener en cuenta los intereses de una persona, sino de varias en conjunto. Es necesario estudiar cómo compatibilizar las preferencias de las partes, priorizando ciertas consideraciones al hacerlo. Y es aquí donde se abre una multitud de nuevas líneas de investigación con aplicación inmediata, que habitualmente están enfocadas al ocio (restaurantes, cine, música, etcétera) aunque las hay igualmente para actividades de formación e información. Y al hilo de esta propuesta surge una interesante línea de investigación relacionado con las Redes Sociales, ya que los recomendadores grupales pueden utilizar estas Redes como plataforma de ejecución y/o aprovechar la información implícita existente en ellas entre usuarios para mejorar su rendimiento.

En este proyecto hemos desarrollado un recomendador grupal de películas conectado a la red social *Facebook*. Proponemos una arquitectura de Razonamiento Basado en Casos (CBR: *Case Base Reasoning*) distribuida, donde la recomendación final se verá influenciada por la personalidad de cada componente y la forma en la que están conectados a través de sus relaciones sociales, básicamente de amistad, que se definen en la red social. Así pues, mediante el CBR, podemos ofrecer a un grupo un producto que se ajuste al compendio de las necesidades individuales con tal de conseguir la máxima satisfacción. Como hemos comentado, nos dedicaremos a la recomendación de películas, pero los métodos utilizados, se podrían integrar en cualquier Red Social, no necesariamente *Facebook*, para proporcionar capacidades de recomendación a los grupos de usuarios para actividades como la elección de un restaurante, planificar un viaje... En nuestras recomendaciones usaremos técnicas existentes de filtrado colaborativo, teniendo en cuenta la personalidad del grupo y las conexiones sociales individuales del mismo, con la que predeciremos la película que debería desear el grupo como entidad.

Es común que los usuarios de las Redes Sociales creen eventos y que entre los amigos se inviten a actividades para realizar en grupo. Es también habitual que los usuarios participen en el evento y ya luego ultimen los detalles (ir al cine, al restaurante...). Esta aplicación, que hemos integrado con la nueva Api de *Facebook: Facebook Open Graph*¹, tiene como primera intención que el usuario la entienda y la utilice como si de un evento normal de *Facebook* se tratase. Lógicamente el fin del evento será el ir a ver una película todo el grupo de amigos, y lo que queda por decidir es qué película se va a ir a ver. Aquí partimos con cierta ventaja con respecto a los eventos normales de *Facebook*, ya que no requiere de una comunicación continua y a veces eterna entre los usuarios para decidir la película. Esta decisión pasará a ser automatizada tras un paso de modelación de las preferencias de cada usuario, y más tarde, la agregación de estas preferencias para obtener la recomen-

¹<http://developers.facebook.com/>

dación del grupo. Se trata pues de eliminar situaciones de conflicto entre distintos integrantes de un grupo. Al estarse ejecutando en la Red Social, podemos inferir algunos aspectos sociales sobre los usuarios. La motivación social hace que los datos que aparecen en sus perfiles (nombre, edad, género, estado civil, localización, e incluso listados de productos, actividades o lugares que gustan) sean consistentes, y como suelen usar de forma frecuente la Red es probable que estos estén actualizados. En nuestro recomendador, utilizamos la personalidad para medir el grado de permisividad del usuario cuando sus elementos preferidos no sean seleccionados por el grupo. Y el nivel de confianza social es aprovechado para calcular cómo podrían influir las preferencias de los amigos en la elección del usuario.

1.2. Objetivos

El objetivo principal de este proyecto ha sido crear una aplicación que fuese accesible a una gran cantidad de personas, para poder simular de la manera más realista posible el proceso de argumentación seguido por grupos de personas al coincidir en una actividad común. La arquitectura está basada en jCOLIBRI (Díaz-Agudo et al., 2007). Nuestro método de recomendación y la arquitectura de nuestra aplicación de *Facebook* son válidos para todos dominios con los productos valorados (comidas, películas, música...). Sin embargo, los hemos aplicado a un dominio específico, el de las películas, y hemos creado *HappyMovie*.

Pretendemos, por un lado, aportar una visión general de la aplicación que hemos desarrollado y demostrar que las recomendaciones a grupos de personas conectados a través de las estructuras de la Red Social mejoran los métodos existentes en la actualidad. Por otro lado, se ha buscado que esta aplicación permita continuar investigaciones futuras en la línea de este trabajo y de su experimentación, así como extender las recomendaciones del grupo a diferentes dominios. Por ello se ha optado por desarrollar la aplicación en la red social *Facebook*, donde podemos extraer una gran cantidad de información de los perfiles de los usuarios.

Para Happy Movie se plantean las siguientes expectativas:

- Reflejo de las interacciones reales entre usuarios cuando se desarrolla un proceso de recomendación mediante la inclusión de factores sociales extraídos a través de *Facebook*. Ejemplos de estos factores sociales son la distancia en la red social, el número de amigos comunes, la intensidad, la intimidad y la duración de la relación, la distancia social, y el porcentaje de fotografías donde aparecen juntos, entre otros. Aunque

como discutiremos en el Capítulo 3, el factor más relevante es el número de amigos en común, seguido por las imágenes, la duración, los intereses comunes y el estado.

- Valoración de la influencia de los distintos perfiles de personalidad de cada usuario involucrado en el proceso.
- Mejora de las técnicas existentes que se usan típicamente a la hora de recomendar distintos tipos de productos a grupos mediante la agregación de los ratings (valoraciones o calificaciones de un producto) para individuos.
- Inclusión de interacciones. Es decir, gestión de información personal, para realizar las recomendaciones, sobre el comportamiento en situaciones de conflicto para cada uno de los miembros del grupo, e información de cómo interactuarán entre ellos.
- Gestión de la personalidad de cada individuo. Es decir, la forma en la que reaccionará cuando el sistema recomienda una propuesta diferente a la que él quería.
- Inclusión de un sistema con memoria para evitar repeticiones en las recomendaciones y un control sobre la felicidad de los usuarios con las recomendaciones para asegurar una satisfacción homogénea.
- Gestión automática de la cartelera de cine. Conexión a un sitio web donde actualice la cartelera. Son necesarias las fichas de las películas para rellenar los campos que se comparan en el recomendador individual.
- Lectura de las conexiones sociales entre usuarios.
- Interfaz gráfico intuitivo. Se busca que los usuarios no necesiten un periodo de aprendizaje para el manejo de la aplicación.

Para lograr satisfactoriamente estos objetivos ha sido necesario previamente realizar una amplia revisión del estado actual de la investigación en sistemas de recomendación. A partir de ella hemos podido caracterizar nuestro sistema de recomendación, detectando debilidades y a su vez proponiendo posibles líneas de mejora y extensión. Esta aplicación está diseñada para la recomendación de películas, pero *HappyMovie* es independiente del dominio y se podrán incluir recomendadores de otros productos como viajes, libros, restaurantes, música... También se ha efectuado una amplia recopilación bibliográfica de los productos software que se han empleado para la creación del proyecto.

1.3. Estructura de la memoria

El contenido de esta memoria está organizado de la siguiente manera:

- En el Capítulo 2 describimos el estado del arte de los sistemas de recomendación para individuos, los sistemas de recomendación para grupos, y para la Redes Sociales, centrándonos en *Facebook* y los sistemas de recomendación y aplicaciones en dicha Red. Dado que nuestro modelo se basa en estos sistemas, hemos realizado una revisión previa sobre el estado del arte de los mismos con el fin de conocer su estado actual, sus ventajas e inconvenientes. Nuestra intención es sacar los beneficios y las buenas ideas presentes en las soluciones actuales y paliar ciertas carencias he hemos detectado en los mismos. En este capítulo también se engloban las tecnologías usadas.
- Después, en el Capítulo 3 presentamos nuestra aplicación como modelo de integración del sistema de recomendación en la red social *Facebook*, también se detalla la implementación y la descripción funcional del sistema.
- Para finalizar, en el Capítulo 4 enumeramos las conclusiones a las que hemos llegado al realizar el trabajo, así como las líneas de trabajo futuras que puede tener nuestro proyecto.

Capítulo 2

Estado del Arte

2.1. Sistemas Recomendadores

Los sistemas recomendadores nacen como una herramienta de ayuda para los usuarios de Internet. Antes, el consumidor tenía un acceso limitado a la información referente a cualquier producto u opción existente en la Red. La publicidad era el medio principal y prácticamente único para dar a conocer un producto, y el problema del usuario era como conseguir una información veraz. Ahora la situación ha cambiado radicalmente. De la escasez de información se ha pasado a la saturación. De tener posiblemente muchas estanterías llenas de CDs y vídeos, a disfrutar de acceso a una cantidad ingente y casi inagotable de creaciones culturales en tiendas online o en redes P2P. Ahora el problema se ha convertido en como separar lo que queremos de lo que no queremos encontrar. Aquí es donde juega un papel importante los sistemas de recomendación. Nacen con el propósito de facilitar la toma de decisiones en temas o dominios en los que las posibilidades existentes son muchas y muy variadas. Esta herramienta ayuda al usuario a obtener la información que necesita de acuerdo a sus preferencias. Luego, toma decisiones basándose sólo en la información disponible.

En la actualidad existen multitud de recomendadores para todo tipo de productos: películas, música, restaurantes, viajes, noticias, etcétera. Pero si cabe destacar un campo de gran aplicación para estos sistemas es sin duda todos aquellos relacionados con el ocio. Muchas de las diferentes actividades que desarrollamos cotidianamente suelen realizarse en grupo, como ver una película, ir a un restaurante, viajar o escuchar una emisora de radio. Luego tiene sentido no solo realizar recomendaciones individuales sino también realizar sugerencias a todo un grupo de personas en función de las preferencias individuales de sus miembros.

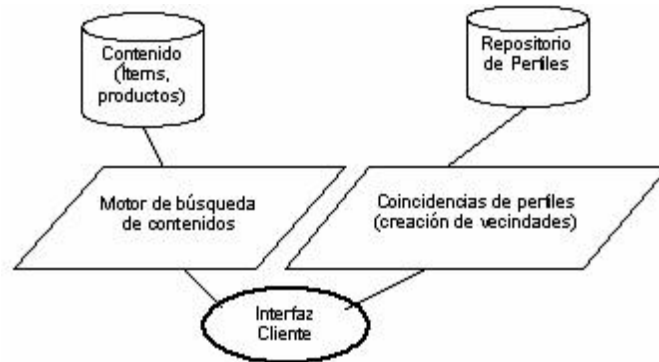


Figura 2.1: Esquema de funcionamiento básico de un Sistema Recomendador

A continuación explicamos los tipos de sistemas recomendadores, sus diferencias y las herramientas que existen para implementarlos.

2.1.1. Recomendadores individuales

Estos sistemas tratan de conocer al cliente para poder ofrecerle algún producto que muy posiblemente podría estar interesado; no como una acción complementaria al marketing, sino como un servicio más adaptado a sus preferencias reales, y por tanto con mayores posibilidades de éxito. Son sistemas que utilizan bases de datos que recogen, procesan y almacenan información del cliente, bien la que éste nos suministra o la que logramos deducir de su comportamiento.

Existen distintas técnicas que pueden ser usadas por los recomendadores para presentar aquellos productos que pueden interesar al usuario. Depende, principalmente, de la forma en que se realice el filtrado de la información. Según dicen los autores de (Nichols, 1997), (Malone y Cohen, 1987) son descritas tres formas de realizar el filtrado de información: social (colaborativo), de contenido (cognitivo) y económico. El filtrado colaborativo se basa en grandes cantidades de calificaciones explícitas por parte de los usuarios. El filtrado de contenido es más cercano al enfoque tradicional de recuperación de información teniendo en cuenta palabras claves. El filtrado económico se basa en ciertos factores de costos asociados como pueden ser el costo/beneficio del uso de la información, el ancho de banda disponible y el tamaño mínimos de los objetos, el uso de este filtrado es muy reducido en sistemas actuales.

Una vez dicho esto, hay que aclarar que entre las dos grandes familias de recomendadores, los basados en contenidos y los de filtrado colaborati-

vo, existen múltiples aproximaciones híbridas que mezclan características de ambas.

2.1.1.1. Sistemas Recomendadores colaborativos

La gran mayoría de los sistemas de recomendación trabajan con enfoques de filtrados colaborativo. En términos simples, por filtrado colaborativo se entiende al método de hacer predicciones automáticas acerca de los intereses de un usuario, colectando información sobre los gustos de varios usuarios (colaboración). Es decir, los sistemas recomendadores colaborativos mapean usuarios con intereses similares y luego crea recomendaciones sobre esa base. Sugiere nuevos productos para un usuario particular basado en los gustos previos del usuario y en las preferencias de otros usuarios que manifiestan gustos parecidos, y el sistema va refinándose progresivamente a medida que mostramos reacciones a las sucesivas recomendaciones. Lógicamente el sistema ofrece la posibilidad de que nos ponga en contacto con productos que o bien no conocíamos, o no nos había llamado particularmente la atención, pero que podría llegar a interesarnos. El sistema no necesita solicitar al usuario una gran cantidad de información, de manera que resulta poco intrusivo. Se limita a incorporar información a medida que el usuario elige opciones de visualización, solicita información de productos, de manera que la página web “aprende” de ese usuario y se adapta a él.

Uno de los algoritmos más usados es el *Nearest Neighbourhood* (Acercamiento al Vecino más Cercano) (Dasarathy et al., 2000). Se utiliza este algoritmo para chequear por similitud a los usuarios con respecto al usuario actual, es decir, el usuario al que hay que recomendar. Y combina las preferencias para determinar una predicción o lista de recomendaciones para este usuario. En una red social, un usuario particular es un vecino con gustos e interés similares que pueden ser encontrados calculándolos con la correlación de Pearson, recolectando las preferencias de los primeros N-vecinos para un usuario en particular (pesado por similitud), y se predice la preferencia del usuario usando ciertas técnicas. Entre sus inconvenientes se encuentra la necesidad de disponer de un número mínimo de usuarios con un número mínimo de predicciones cada uno, incluido el usuario para el que se pretende realizar la recomendación. Otras formas de calcular esa vecindad o similitud en gustos e intereses es mediante vectores (medida del coseno), las medidas de correlación basadas en entropía, correlación de Ringo (Shardanand y Maes, 1995) o correlación de Spearman (Herlocker y Riedl, 2002).

Existen dos formas de recoger estas valoraciones. Una es de forma explícita, es decir el usuario asigna una puntuación a cada elemento que será un valor numérico discreto entre un máximo y un mínimo. La segunda forma

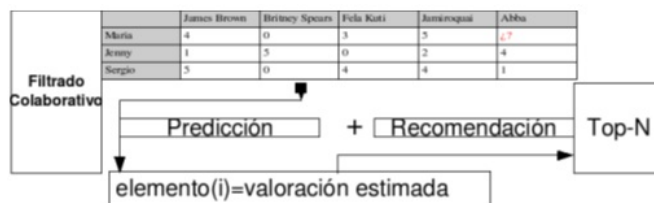


Figura 2.2: Modelo de un sistema de filtrado colaborativo

es recoger las valoraciones implícitamente, extrayendo la información pertinente de las acciones del usuario. Por ejemplo el tiempo que pasa leyendo una determinada página web, los enlaces que sigue, el número de veces que se escucha una canción, esta sería una aproximación más clásica de minería de datos. Una vez que se tiene suficiente información del usuario se pasa a la fase de predicción y recomendación. Predicción hace referencia a estimar que valoración daría a cada elemento mientras que recomendación se refiere a extraer los n elementos más recomendables.

En la actualidad el desarrollo del filtrado colaborativo se encuentra muy activo y genera un gran número de publicaciones y congresos todos los años. Algunos ejemplos actuales son los recomendadores en tiendas on-line. La web pionera de este tipo de recomendadores fue Amazon¹. Partiendo de un producto se recomiendan otros que han interesado a los usuarios que compraron dicho producto. También podríamos mencionar recomendadores musicales, de libros, de películas. En los últimos años han surgido decenas de webs de este tipo entre las que destacan Last.fm² y MyStrands, esta última de origen español. En estos servicios, cada vez que un usuario escucha una canción se envía su información a la base de datos del sistema, el cual las utiliza para generar nuestras recomendaciones, pero las funcionalidades que ofrecen crecen constantemente. Por ejemplo Last.fm ofrece radios personalizadas para cada usuario en función de las recomendaciones que reciba, y MyStrands organiza fiestas en las que la música se elige automáticamente de forma colaborativa en función de los gustos de los asistentes. El modelo de negocio de estas empresas, es además de la publicidad, el de acuerdos con tiendas on-line para enlazar directamente las recomendaciones con su servicio de venta. También el de proporcionar a las compañías análisis de tendencias musicales, de nuevos artistas... Otro ejemplo de uso es la búsqueda de personas afines en comunidades. En webs como meneame.net se tienen en cuenta las noticias que cada usuario ha votado para generar una lista de vecinos con similares intereses.

¹<http://www.amazon.com>

²<http://www.last.fm>

En contraposición a los recomendadores colaborativos, encontramos los recomendadores basados en contenidos, que pasamos a describir a continuación.

2.1.1.2. Sistemas Recomendadores basados en contenidos

Los sistemas recomendadores basados en contenido sugieren productos que son similares a aquellos que previamente valoró el mismo usuario. La idea que subyace es que si te gustaron esos productos, es probable que te gusten otros similares. Estos productos son definidos según sus características generando así descriptores de los productos. El perfil de un usuario se basa en cómo valora esas características. Por consiguiente, es recomendable usar estos sistemas con productos basados en la correlación entre el contenido de los productos y las preferencias de los usuarios.

Una recomendación basada en contenido tiene como ventajas que no da pie a opiniones subjetivas de otros usuarios, y la recomendación sigue una ruta que hizo en base al historial del usuario, sirviendo esto luego como explicación de la recomendación dada. Por contra, está muy limitado al recomendar sólo productos similares a los que ya recomendó. Presenta dificultades en dominios con contenido difícil de analizar (música, vídeos, gráficos...). El usuario tiene que valorar un número suficiente de productos para que el sistema pueda realmente entender sus preferencias. Y además, es muy difícil para el sistema aprender a adaptarse a los cambios en el perfil del usuario hasta no haber recolectado un número suficiente de ratings actualizados. Como ejemplo de recomendador basado en contenido podemos mencionar a Pocket Restaurant Finder, que recomienda restaurantes para grupos de usuarios basado en la localización y las características culinarias del restaurante.

Los recomendadores basados en casos son un tipo especial de recomendadores basados en contenidos, en los que cada caso que se recupera representa una recomendación anterior de un producto similar al que el usuario desea. Tiene un origen común con las técnicas de razonamiento basado en casos (CBR: *Case-based Reasoning*). Los sistemas CBR (Leake, 1996) cuentan con una base de casos que se compone de problemas ya resueltos, junto con la solución tomada. De este modo los nuevos problemas se resuelven adaptando soluciones pasadas que fueron usadas para resolver problemas similares. Así que podemos decir que a favor del uso de este tipo de técnicas es que propone soluciones rápidamente. No es necesario conocer completamente el dominio. Los casos son muy útiles para conceptos mal definidos. Y que son resaltadas las características más importantes. En contra hay que decir, que los casos viejos pueden ser pobres. Los más apropiados pueden no ser recuperados. Y

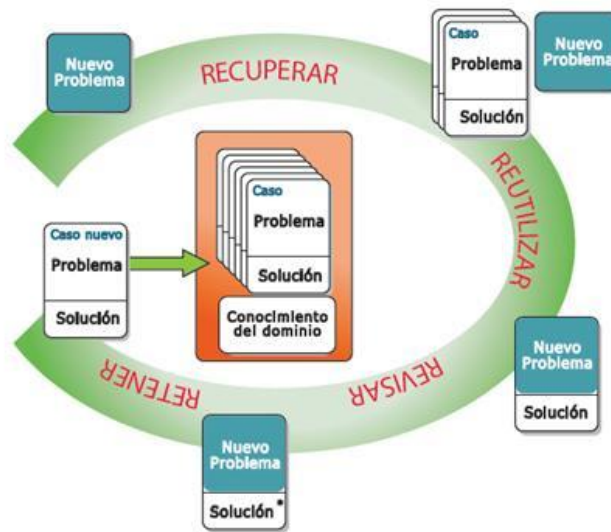


Figura 2.3: Fases del ciclo CBR

es posible que necesite conocimiento para realizar la adaptación. Estos sistemas basados en casos completan un ciclo de razonamiento de cuatro etapas:

- Recuperación. Selecciona los casos importantes para el problema que se plantea
- Reutilización. Adapta la solución de los casos al problema actual
- Revisión. Un experto examina la solución sugerida, y determina si es correcta
- Recordar o Retener. El sistema decide si incorpora la solución en la base de casos

Este tipo de recomendadores se pueden clasificar atendiendo a ciertas características (nótese que algunas de estas características también pueden encontrarse en los recomendadores colaborativos). A continuación pasamos a comentar las características de estos sistemas.

2.1.1.3. Características de los Sistemas de Recomendación

Las características de los sistemas de recomendación son:

- Quién toma la iniciativa. Podemos distinguir dos características en base a quién lleva la iniciativa en la recomendación. Así, podemos tener un

recomendador reactivo, donde es el usuario quien lleva la iniciativa realizando una consulta al sistema. Por otro lado está el recomendador proactivo, donde el que lleva la iniciativa es el recomendador, realizando una propuesta inicial al usuario basada en el historial pasado del usuario, o en valoraciones asociadas a los elementos, o en cualquier otra estrategia previamente seleccionada.

- **Dinámica de recomendación.** También aquí distinguimos dos características: *single-shot* y *conversacional*. *Single-shot* son aquellos en los que sólo se muestra un conjunto de elementos recomendados al usuario y este tiene la oportunidad de elegir uno o descartarlos. Si la recomendación no agradara al usuario, este debería empezar de nuevo para obtener nuevos elementos. Los recomendadores conversacionales son aquellos en los que la recomendación se entiende como un proceso iterativo, en el que el usuario puede ir refinando sus requisitos hasta obtener un elemento adecuado para él. Existen dos estrategias de conversación. *Navegación-por-propuesta* y *navegación-por-pregunta* (Shimazu, 2002)(Shimazu, 2001). En la primera, un conjunto de elementos es mostrado al usuario, a partir del cual éste podrá refinar sus requisitos. En la *navegación-por-preguntas* el sistema recoge los requisitos del usuario a partir de un conjunto de preguntas cuidadosamente seleccionadas.
- **Capacidad de personalización.** Es decir, si en el proceso de recuperación intervienen o no las características/preferencias/necesidades del usuario. Los perfiles de usuario pueden contener información sobre el historial de navegación, las preferencias, las necesidades del usuario, o lo que se crea conveniente. La capacidad de personalización está relacionada con cómo se maneja toda esta información, es decir, si es tenida en cuenta o no a la hora de realizar las recomendaciones. También hay que tener en cuenta las técnicas utilizadas para mantener actualizados los perfiles y la persistencia de los mismos.
- **Determinación de la calidad de los elementos.** Recomendadores que siguen una aproximación tradicional de la similitud entre elementos, o recomendadores que apuestan por innovar en la similitud introduciendo una medida de calidad en los elementos (McGinty y Smyth, 2003). Esta medida está relacionada con la diversidad de los elementos recuperados en la recomendación. Se define la diversidad del conjunto recuperado como la disimilitud existente entre cada par de elementos del conjunto. Es decir, un elemento mejorará su calidad cuanto más disimilar sea a los ya recuperados, siempre y cuando siga manteniendo la similitud con la consulta.

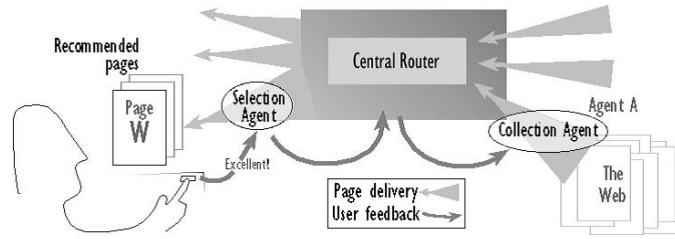


Figura 2.4: Descripción de la arquitectura Fab

2.1.1.4. Sistemas Recomendadores híbridos

Antes, hicimos mención a los sistemas híbridos, que son aquellos que combinan tanto la técnica basada en contenidos como la de filtrado colaborativo, reduciendo así las limitaciones de cada una de ellas. Es por ello que puede usar las experiencias de otros, en vez de un análisis impreciso e incompleto de contenidos (ventaja correspondiente a la colaborativa). Puede tratar productos no vistos por otros (ventaja correspondiente al basado en contenidos). Permite hacer buenas recomendaciones a usuarios, usando los perfiles contruidos a partir del contenido de los productos, aun si no hubiera usuarios similares. Y los usuarios que no han calificado los mismos productos se benefician de recomendaciones colaborativas.

Un ejemplo de todo esto sería el sistema *Fab* (Malone et al., 1987), un sistema híbrido que recomienda al usuario páginas Web. El sistema modela el perfil del usuario basado en el contenido de los análisis (cuando un usuario califica una página) y compara estos perfiles para determinar similitudes entre usuarios para una recomendación colaborativa. De esta manera, el usuario recibirá páginas, tanto las que ha calificado relevantes (con respecto a su perfil) como las que han recibido calificaciones altas por usuarios con un perfil similar al suyo (vecinos cercanos). La arquitectura de *Fab* se encuentra formada por tres componentes principales: agentes de colección, que seleccionan páginas de un tema específico; agentes de selección, los cuales encuentran páginas para un usuario específico; y un “ruteador” central que los conecta. Cada agente mantiene un perfil del usuario basado en las palabras que contienen las páginas Web que el usuario ha calificado, como se muestra en la Figura 2.4.

A continuación pasamos a explicar las diferencias entre los recomendadores individuales y los grupales.

2.1.2. Recomendadores grupales

Los sistemas recomendadores han estado dirigidos tradicionalmente a realizar recomendaciones individuales. Sólo en los últimos años se han empezado a desarrollar técnicas que permitan hacer recomendaciones para grupos de personas. Es un tema de investigación abierto, en el que hay permanentemente nuevas aportaciones.

En general, estos sistemas se ocupan de encontrar una estrategia para, o bien agregar las preferencias individuales de cada miembro, o bien combinar modelos de usuarios individuales para obtener la preferencia del grupo. Por otro lado, la mayoría de estos sistemas han optado por aplicar métodos para la adquisición de información casi idénticos a los utilizados por los recomendadores individuales, haciendo caso omiso de la personalidad de cada miembro y las relaciones entre los usuarios dentro del grupo. Del mismo modo, podemos distinguir entre dos métodos para la recopilación del conocimiento del usuario:

- *Explícitos*. En este tipo de recomendadores para grupos, requiere que cada usuario le “diga” al sistema de manera explícita sus preferencias respecto a la información de los productos. Podemos hacer una diferenciación entre dos tipos: aquellos en los que el usuario califica categorías y los que califican productos. Un ejemplo del primero es el recomendador de música MusicFX (McCarthy y Anagnost, 1998), donde cada usuario debe calificar su interés sobre cada género musical. En Polylens (O’Connor et al., 2001), que es un recomendador de películas, el usuario en vez de calificar categorías califica directamente películas. Es una extensión del sistema *MovieLens*³, basado en técnicas de filtrado colaborativo, donde el usuario se le otorga una escala para puntuar películas. No es una recomendación explícita, pero este sistema consigue hacer recomendaciones a grupos agregando las preferencias de los miembros del grupo con el fin de minimizar el ‘malestar’, basándose en la hipótesis de que la felicidad del grupo será igual a la del menos feliz de sus miembros.
- *Implícitos*. Muchos sistemas recomendadores no requieren que sus usuarios especifiquen de manera explícita sus preferencias. Simplemente, se registran las acciones de los usuarios y se concluyen las calificaciones sobre estas acciones. Un ejemplo es el sistema FIT (Miller y Kenyon, 2007), que es un modelo no intrusivo para el modelado de los patrones de exposición al televisor de por ejemplo toda una familia. FIT solicita a los usuarios que ingresen sus datos personales y la hora actual,

³<http://www.movielens.org>

y además deben especificar las horas en las que prefieren ver la televisión. A partir de esta información, FIT infiere qué subconjunto de los miembros de la familia estará frente al televisor y es capaz de sugerirle programas. Otro ejemplo es Let's Browse (Lieberman et al., 1999), que recomienda páginas web a grupos que están navegando en la red. Mientras navegan, el sistema va analizando las palabras que aparecen en las páginas visitadas por el grupo.

Se dan casos donde es bueno tener en cuenta el foco en preferencias negativas, donde el recomendador sugiera basándose en evitar aquello que pueda disgustar a algún miembro del grupo. Un ejemplo de ello es el sistema Adaptive Radio (Chao et al., 2005). Otros sistemas se plantean el compartir información sobre determinadas preferencias para ampliar su espectro de intereses, con las ventajas de ahorro de esfuerzo al ingresar los datos propios y de aprender de otros miembros basándose en sus experiencias o conocimientos. Tavel Decision Forum (Jameson, 2004) vale como ejemplo de sistema que permite observar las preferencias de los otros usuarios.

El conocimiento de este tipo de información permite persuadir a otros miembros para que especifiquen una preferencia similar. O bien sirve para exponer cierta preferencia que afectará a la decisión final del grupo aunque no sea una preferencia generalizada. Es decir, supongamos un recomendador para seleccionar un restaurante para comer. Un miembro podría señalar que es celíaco para evitar restaurantes tales como pizzerías. El conocimiento de esa información por parte de los demás integrantes haría que ese tipo de recomendaciones se descartasen, aun el caso que por otros medios tuviera una calificación para el grupo muy alta. También un punto a sopesar es la privacidad. Como acabamos de ver, tener acceso a las preferencias de un miembro puede tener muchas ventajas. Sin embargo, puede darse el caso, por cualquier motivo, que cierta información pueda hacerle sentir avergonzado.

Es obvio darse cuenta que estos sistemas son fácilmente manipulables. Esta manipulación se da cuando el usuario califica muy positiva o negativamente un producto para asegurarse así su salida o no en la recomendación. Se conoce como actuación desleal hacia el grupo, y una versión temprana de MusicFX utilizaba una fórmula de agregación de preferencias fácilmente manipulable. Existen mecanismos para evitar la manipulación, como la realizada por Tavel Decision Forum promediando las calificaciones.

Por norma general, los recomendadores acaban escogiendo aquella opción que satisface al mayor número de usuarios del grupo, de acuerdo con las preferencias individuales. Los motivos pueden ser varios: el tamaño del grupo, cierta heterogeneidad dentro del mismo... Para intentar no caer en esto, son

necesarios buenos métodos de agregación para poder combinar la información sobre las preferencias individuales de los usuarios y obtener una “afortunada” recomendación para el grupo. Atendiendo a la teoría (el resultado del teorema de Arrow (Dopico, 2007)) nos dice que no existe ninguna regla de agregación de preferencias que tenga las propiedades normativas deseables (los criterios racionales: transitividad, completitud y reflexividad, y los principios de no-dictadura y de no-imposición) salvo que las preferencias sean el fiel reflejo de las preferencias de algún individuo, denominado “dictador”. Por este motivo debemos buscar la mejor aproximación, surgiendo así tres aproximaciones que tratan de resolver este problema:

- Mezclar las recomendaciones que se harían por separado a cada uno de los miembros del grupo. Es un método de agregación de las valoraciones individuales de cada componente del grupo para formar una única lista. Este método fue considerado por PolyLens y para ello aplican la estrategia “Least Misery” (minimización de la miseria) (Jameson y Smyth, 2007).
- Agregar las valoraciones para cada usuario. Para cada producto candidato y para cada miembro del grupo, el sistema predice cómo ese componente evaluaría dicho producto, y devuelve una colección de candidatos que tengan las valoraciones previstas más altas. Un ejemplo de esto es el Pocket Restaurant Finder (McCarthy, 2002).
- Construir un modelo de las preferencias del grupo. El sistema realiza una combinación lineal de los modelos individuales de los componentes del grupo para construir un modelo de preferencias para el grupo. Let’s Browse es un ejemplo de este tipo de sistemas.

2.1.2.1. Principales estrategias de agregación de calificaciones

Como hemos visto, los recomendadores en general se ocupan de encontrar una estrategia para agregar las preferencias individuales o combinar modelos de usuarios individuales para obtener la preferencia del grupo. Esto es de especial interés puesto que el grupo puede ir cambiando al sumarse nuevos integrantes, o no estar presente alguno de sus miembros.

Una forma para obtener la preferencia del grupo podría consistir en hacer que sus miembros votasen. De este modo, el acuerdo obtenido estaría basado en las preferencias individuales. El estudio de agregación de preferencias tiene como antecesor el estudio de los sistemas de votación. Atendiendo a las características de este último podemos describir distintas estrategias para la agregación de calificaciones, que es fundamental para la realización de adecuadas recomendaciones para grupos. Por ejemplo, habrá momentos que esa recomendación deba resultar de una voluntad consensuada por todos los

miembros, pero pueden existir circunstancias donde sea más adecuado tomar en cuenta con mayor peso la opinión de un integrante.

Al hablar de los sistemas de votación y de la estrategia a seguir podemos recurrir a los trabajos realizados por Masthoff que ilustran diversas estrategias, provenientes de la teoría social de las elecciones (*Social Choice Theory*). Nos ayuda a establecer las propiedades deseables con las que debe contar el sistema de votación, como son: la *votación anónima o simétrica*: si cualquier permutación de preferencias entre individuos no altera la ordenación social de alternativas; la *asociación positiva*: si la alternativa x precede a y en la lista de productos final, entonces un incremento en el rating de x por parte de un individuo debería mantener la posición de x anterior a y en la nueva lista final. Si la alternativa x aparece igual a la alternativa y en la secuencia, entonces un incremento en el rating de x por parte de un individuo debería hacer que x preceda o siga igual a y ; el *criterio del ganador de Condorcet*: establece comparaciones uno contra uno y comprueba si existe un candidato que, tales enfrentamientos, gane todos. En tal caso, ese candidato sería el elegido. Este sistema de elección tiene el inconveniente de que se pueden producir situaciones cíclicas y no siempre existe ganador; la *satisfacción de cada individuo con el resultado debe estar por encima de determinado umbral*: que pretende que el producto elegido no disguste a ninguno del grupo. Es decir, evita aquellos productos que pueda disgustar a alguno de los miembros. Este criterio es natural, porque a ningún integrante del grupo le gusta que otro esté extremadamente disgustado con lo que decidieron el resto del grupo; o la *regla del Pareto*: si al menos una persona prefiere x a y y nadie prefiere y a x , entonces x debería estar sobre y en el ranking. Si nadie prefiere x a y y nadie prefiere y a x , entonces x e y deberían estar en el mismo lugar del ranking. Ha sido probado que no todos los sistemas pueden tener todas estas propiedades simultáneamente, y que para cualquier método de agregado de preferencias individuales para obtener la opción del grupo, es posible encontrar patrones de preferencia individuales que den lugar a un patrón de elección social, que no es una ordenación lineal, lo que constituye presumiblemente, el motivo de que en distintos países existan distintos métodos de votación.

Es necesario establecer qué procedimiento computacional se va usar para la agregación de calificaciones. Estos procedimientos o estrategias permiten agregar o fusionar las calificaciones (ratings) o los grados de interés expresados por individuos por determinados productos, para obtener el grado de interés predicho para el grupo que conforman dichos individuos. Existen puntos que se han de procurar cubrir a la hora de optar por un determinado método, la satisfacción total o el grado de igualdad son dos claros ejemplos. Atendiendo a una de estas técnicas o métodos, cada uno de los miembros

debe votar su alternativa preferida y la que obtenga mayor número de votos gana, mientras que en otras cada usuario puede votar por tantas alternativas como desee. Otros enfoques hacen una nueva lista de puntuaciones con el mínimo (o máximo) de las calificaciones individuales. Sin embargo, a partir de los experimentos descritos en Masthoff (Masthoff y Gatt, 2006), se deduce que la mejor técnica de fusión de preferencias en cuanto a la exactitud de la recomendación para el grupo es la que se nombra Promedio Sin Miseria (Average Without Misery), donde la lista de puntuaciones se obtiene realizando un nuevo promedio de las calificaciones individuales, pero sin tener en cuenta los valores inferiores a un umbral determinado. De esta forma, se evita recomendar productos que disgusten a algunos miembros del grupo.

A continuación exponemos algunas de las estrategias con un poco más de detalle:

- Regla de Copeland. Esta es una forma de votación por mayoría. Se ordenan las alternativas según el índice Copeland, es decir, según el número de veces que una alternativa gana a otras (tiene mayor calificación) menos el número de veces que pierde con otras.
- Estrategia de Minimizar la Miseria. Aunque la satisfacción media sea alta, si una solución deja a un componente del grupo especialmente disconforme, se considera como una situación indeseada. La calificación de grupo para un producto será como resultado de la calificación mínima otorgadas en los perfiles de los miembros del grupo. Por lo tanto, un grupo estará tan satisfecho como su miembro menos satisfecho. PolyLens utiliza esta estrategia, asumiendo que son pequeños los grupos de personas que ven una película en conjunto, y por lo tanto, influye mucho que un miembro esté a disgusto. Una desventaja es que la opinión de una minoría puede disponer lo que verá el grupo.
- Estrategia de Maximización de Satisfacción. La calificación del grupo para cada producto se obtiene de los máximos de las calificaciones individuales. Aquel producto que haya obtenido la calificación resultante más alta, aparecerá antes en la secuencia, o se asume como preferido del grupo. Esta estrategia se asegura que el producto propuesto sea de gran interés al menos para una parte del grupo.
- Estrategia de Promedio sin Miseria. Se realiza una nueva lista con los promedios de las calificaciones individuales, pero sin contar con aquellos productos que han sido valorados por debajo de cierto umbral. De este modo, se elige un contenido que despierta la satisfacción promedio más alta, siempre que no disgusten en demasía a ningún miembro del grupo.

- Estrategia de Equidad. Se seleccionan los productos preferidos de cada uno de los individuos. En el caso de que distintos elementos tengan la misma calificación, la opinión de otros individuos es tomada en cuenta. La idea que subyace en esta estrategia es que no es tan malo para un individuo un cierto producto que no le gusta, siempre y cuando se acabe recomendando aquellos que realmente le satisface. Esta estrategia es utilizada normalmente cuando un grupo intenta dividir equitativamente una serie de elementos: una persona elige primero, luego la siguiente, hasta que todos hayan hecho una elección. Luego todos eligen una segunda alternativa, a menudo empezando con la persona que eligió último en la ronda anterior. Continúa hasta que todas las alternativas hayan sido usadas.

Estas técnicas de agregación han sido empleadas por multitud de recomendadores en sus respectivos dominios. Travel Decision Forum o Intrigue (Ardissono et al., 2003), en la planificación de vacaciones a grupos; Let's Browse, como recomendador de páginas web a un grupo; MusicFX, para ajustar la selección de música en una sala de ejercicio físico de acuerdo a las personas presentes en la sala; y Polynens, que sugiere películas a un grupo de espectadores, por nombrar sólo algunos ejemplos. Hay que decir, que algunas simplemente se basan en ellas para realizar versiones un poco más complejas. Por ejemplo, MusicFX utiliza una estrategia basada en la de Promedio sin Miseria para generar una lista del grupo, dando la posibilidad de eliminar los géneros que estén por debajo de cierta calificación otorgada por cada miembro inferior a un mínimo, para que no haya personas disgustadas.

Para evitar que se reproduzca exactamente el mismo género cada vez que estén las mismas personas, se hace una selección aleatoria ponderada obteniendo una lista con los m géneros mejor calificados, siendo m un parámetro de configuración del sistema, que serán los candidatos a ser reproducidos. Otra posibilidad que llevan a cabo otros recomendadores, son la combinación de las estrategias anteriormente descritas, con el objeto de encontrar el método de recomendación que mejor se adapte a sus necesidades.

Sin embargo, la mayor parte de las estrategias de agregación como las anteriormente mencionadas combinan las preferencias de los usuarios sin tener en cuenta ni las relaciones entre los miembros del grupo ni la relevancia de las preferencias de cada miembro. Se observó que había una necesidad de modificar las estrategias existentes que tienen en cuenta a cada usuario del grupo como igual a los demás.

Trabajos recientes (Recio-García et al., 2009), tratan la mejora de las actuales técnicas de recomendación para grupos mediante la introducción de un factor novedoso: la personalidad de cada individuo en el grupo cuando

se trata de situaciones de conflicto. Utilizan un test de personalidad para obtener los diferentes roles que la gente desempeña al interactuar en una toma de decisiones. Además de la caracterización individual de las personas en el grupo, que también están estudiando otros factores con respecto a la estructura del propio grupo y cómo los usuarios interactúan entre sí. Para la recomendación final, las preferencias de cada individuo tendrán diferentes pesos dependiendo de la manera en que cada miembro del grupo reaccionaría en una situación conflictiva.

La investigación actual ha señalado que la gente tiende a confiar más en las recomendaciones que realizan aquellas personas cercanas y de confianza (amigos), que en las recomendaciones basadas en valoraciones anónimas (Sinha y Swearingen, 2001). Este factor es aún más importante cuando se lleva a cabo una recomendación de grupo donde los usuarios tienen que decidir un tema para todo el grupo. Este tipo de recomendaciones suele seguir un proceso de argumentación, donde cada uno defiende sus preferencias como usuario y refuta las opiniones de otros.

Es lógico pensar que tras una recomendación los miembros del grupo se pregunten cómo se llegó, o como se obtuvo, tal recomendación. Y cuanta “satisfacción” aglutina dicha recomendación para cada uno de los individuos que forman el grupo. Para responder a esta cuestión, en el punto siguiente estudiaremos la aceptabilidad de la recomendación grupal para cada uno de los miembros.

2.1.2.2. Aceptabilidad individual de la recomendación

La recomendación para el grupo debe satisfacer las preferencias de cada miembro para que pueda ser validada y aceptada como recomendación. Por consiguiente se debe estudiar la aceptabilidad de la recomendación grupal para cada individuo, subgrupo y grupo.

Un ejemplo de esto lo encontramos en los sistemas Let’s Browse y Travel Decision Forum. Los usuarios tienen conocimiento cuán de aceptable es la recomendación para los demás miembros del grupo, permitiendo así por ejemplo que una persona pueda dejar a otra contenta con la decisión final. Por su parte, CATS permite que cada usuario pueda criticar la recomendación grupal para obtener una recomendación similar que sea aceptable por él.

Por consiguiente, a la hora de tomar una decisión final no podemos garantizar que se vayan a tomar ninguna de las recomendaciones realizadas. Por ello, algunos sistemas tienden a no proporcionar una única decisión final,

sino un medio para argumentar entre los miembros del grupo y llegar a un consenso.

2.2. Redes sociales

Las “Redes Sociales” han experimentado un auge importante en nuestra sociedad. El estar en contacto con nuestros amigos, conocidos y familiares; compartir con ellos fotos y vídeos, reencontrarse con antiguos compañeros, practicar microblogging, formar parte de fan de clubs, compartir pasión con aquellos que tiene tus mismos intereses, etcétera. Son sólo unos pocos ejemplos de porqué los jóvenes, -y cada vez un mayor número de no tan jóvenes - invierten una considerable parte de su tiempo en el uso de estas redes. La facilidad de acortar distancias que brinda la tecnología, ha producido que las redes sociales sean actualmente el centro de atracción de Internet.

En la actualidad, es completamente necesario adaptarse a las nuevas tecnologías y el uso de estas, pero como en todo es importante saber encontrarle los beneficios y darle un buen uso para que nos aporte algo positivo. Sin embargo aquellas personas o empresas que saben hacer un buen uso de estas redes disponen de una poderosísima herramienta de comunicación con la que compartir ideas, darse a conocer o expandir una marca entre otros.

Una red social, podría definirse como aquel lugar virtual compuesto por una serie de individuos o miembros que les permite una interacción social, el intercambio constante de información, intereses, ideas y opiniones. Es una comunidad en constante construcción y remodelación a fin de ser cada vez más competente frente a las necesidades de los internautas, quienes buscan un sitio apropiado para convivir con demás personas que comparten sus mismos gustos e intereses, pero al mismo tiempo relacionarse con gente, como se muestra en la Figura 2.5.

Las relaciones que pueden darse son de muy distinto tipo y naturaleza. Por ejemplo, de intercambios financieros, amistad, intereses comunes... Y según la naturaleza de las relaciones, las redes sociales se pueden dividir entre:

1. Diádicas (sólo indican ausencia o existencia de la relación) o valoradas (en la que el grado de relación puede cuantificarse y medirse en términos de orden o de peso como, por ejemplo, número de encuentros entre dos personas).
2. Transitivas (la relación en realidad es una unión entre actores con costumbres en común que siempre es recíproca. Ejemplo: leemos el mismo

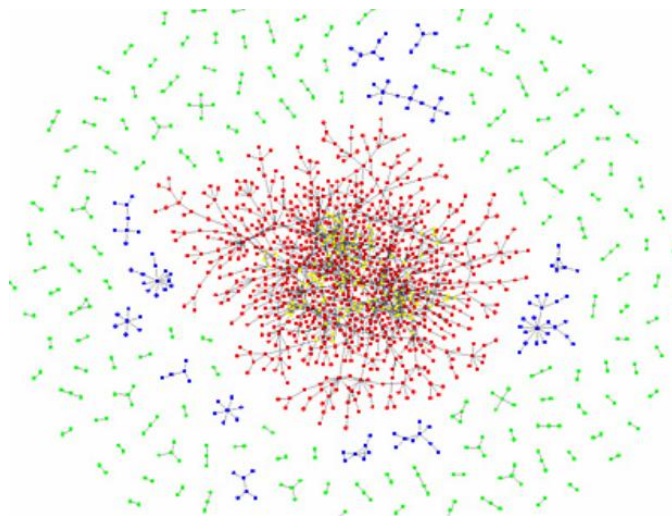


Figura 2.5: Mapa de una comunidad virtual. Distinguimos el núcleo de operación formado por usuarios líderes y regulares de la periferia integrada por el resto

blog habitualmente) o dirigidas (que el individuo A tenga relación con el individuo B no implica que B tenga esa misma relación con A, como, por ejemplo, al prestar dinero).

Actualmente en Internet hay múltiples sitios que implementan una estructura de redes sociales para sus miembros, de esta manera nos podemos encontrar con webs en las que los usuarios se relacionan entre ellos mediante algún elemento común. Como acabamos de afirmar, estos elementos pueden abarcar varias modalidades, ya sea artículos, imágenes, vídeos, intereses comunes, relaciones de amistad... El procedimiento de todas ellas suele ser muy parecido, un miembro invita o agrega a sus preferencias a otros miembros o publica y comparte algún elemento de interés. De esta forma aunque entre dos miembros de una red social no haya una relación directa entre ellos, si pueden llegar a estar conectados indirectamente a través de otros miembros. Lo que hacen las redes sociales en Internet es precisamente materializar los vínculos directos entre los miembros, creando finalmente una gran red en la que cada uno de sus integrantes está de alguna manera vinculado con la totalidad de los otros participantes.

En relación al crecimiento y expansión de las redes sociales, la probabilidad p de que un usuario u entre a formar parte de una determinada red C , depende -entre otros múltiples factores- del número k de contactos del individuo que ya estén en dicha red, tripleta (u, C, k) , siguiendo una tendencia expresada por la relación exponencial (Backstrom et al., 2006):

$$P(k) = a \cdot \log(k) + b$$

donde a y b dependen en gran medida de la posible existencia de relación directa entre los k contactos de u entre sí. Así, la probabilidad de que un usuario u se adhiera a una red social depende no sólo del número de personas que conozca dentro de ella, sino también de cómo están conectadas. Cuanto mayor es el número de conocidos dentro de una red, y más relación existe entre ellos, mayor es la probabilidad de que el usuario se adhiera a dicha red.

A pesar de la existencia de diversas redes sociales, un estudio de la empresa de estadística web Compete Inc.⁴, reconoce a Facebook como la red con mayor tráfico con hasta 200 millones de visitas al mes, seguida por MySpace con 120 millones de visitas, luego Twitter con 74 millones.

Todas las aplicaciones que ofrecen Servicios de Redes Sociales en sitios de Internet tienen una serie de características comunes que les permiten identificarse, tales como:

- Creación de perfiles de usuarios o Social Network Profile(SNP)(Liu, 2007):

Se trata de una sección o página dónde el usuario incluye datos sobre sí mismo. Esta información puede ser privada entre un grupo de personas, semipública o pública según decida su propietario, dependiendo de la casuística y de la política de privacidad establecida por éste.

No obstante, los perfiles de los individuos no sólo sirven para identificarlos sino también para darse a conocer entre los demás, mostrando públicamente a todos los miembros en la red social (directamente conocidos o no) el estatus, rol o imagen a desempeñar y que les distingue en la red, a través de la exposición de sus inquietudes, intereses o preferencias culturales: música, libros, películas, programas televisivos...

En los perfiles aparecen datos identificativos del usuario (su nombre, apodo, cumpleaños, religión...), sociodemográficos (sexo, edad...), geográficos (residencia), intereses e inquietudes personales (ocio, cultura...), rasgos culturales y formativos (por ejemplo, centro de estudios, libros o películas favoritas), a qué se dedica (ocupación), sus razones para encontrarse en la red (qué busca) y un área que el permite escribir libremente acerca de sí mismo. La mayor parte de los sitios de Servicios de Redes Sociales (SNS, Social Network Services) permiten la creación de perfiles con rasgos personalizados, variando la representación o el

⁴<http://www.compete.com>

estilo (colores, formas, texturas) de su sección a su gusto, incluyendo facilidades para la publicación de una fotografía personal representativa, que no tiene por qué ser la del rostro del usuario.

Por otra parte, la descripción de los gustos influye en la percepción de la posición y del prestigio de la persona en la red, representando una oportunidad para la diferenciación dentro de la colectividad, por lo que el individuo suele rellenar estos campos como medio para darse a conocer, asumiendo una serie de roles fijados bajo reglas sociales y la estética a transmitir en determinados grupos, que incluso le llevan a modificar sus preferencias originarias para sentirse alineado e integrado en grupo social con rasgos e intereses comunes.

Sin embargo, la identidad del individuo no queda completamente definida sólo con una descripción atómica de éste, sino que también se hacen necesarios datos acerca de su papel en la red, el grado de interacción y las personas con las que se relaciona. Así, normalmente aparecen datos sobre el grado de actividad o uso de ese miembro en la red (como la última fecha de conexión), información sobre las novedades del usuario en cuanto a sus aportaciones y participaciones (nuevas entradas, comentarios, fotografías, postales, respuestas), o los últimos comentarios recibidos y efectuados dentro de su ámbito de operación. Los listados de amigos y conocidos con los que el usuario guarda relación tienen un papel determinante en lo que respecta al rol del individuo en un determinado grupo social.

- Generación de una lista propia de contactos. Navegación, y operación entre la del resto de éstos:

Existen capacidades para generar vistas agregadas de contactos o enumerar las personas con las que se guarda relación reconocida y consentida dentro de la red, o incluso con las que se pudiera llegar a tener algún tipo de contacto a través de intermediarios. Los usuarios se identifican como amigos entre ellos mediante peticiones formales de anexión a través de mensajes entre pares, que han de ser confirmadas por los extremos.

La exhibición pública de las conexiones con otras personas constituye un medio social para contextualizar la posición y relevancia del individuo en su entorno, con lo que la lista de contactos representa un rasgo que hace referencia a la identidad del usuario. La naturaleza y modo de operación en la consulta y navegación entre los perfiles de conocidos de otros contactos en la red, varía sustancialmente dependiendo

del sitio concreto de redes sociales, no obstante, esta funcionalidad se ofrece de una u otra manera en todos ellos. Las facilidades para listar y consultar los perfiles de otros nodos de la red posibilitan, a través de búsquedas o navegación por éstos, la generación del grafo de la red social de dicho individuo dentro de la comunidad. En muchos casos, esta representación se puede visualizar en las páginas de cada uno de los usuarios, pudiendo efectuar una navegación, a través de los enlaces entre los perfiles que cada persona tiene desde el suyo hacia el de otros.

El establecimiento de enlaces entre nodos de la red, gracias a la consulta de las listas de amigos de intermediarios conocidos, o la creación de enlaces por búsqueda directa, permiten al usuario llegar a ser influyente en su red según el número de contactos de los que disponga, hecho que le conferirá un posicionamiento dentro del grupo y le hará actuar de canalizador, puesto que dispondrá de la capacidad de difundir ideas y tendencias en su comunidad con un mayor potencial. Por tanto, el poder del individuo dentro de una red se puede expresar en términos de enlaces con otras personas, incluso por encima de sus rasgos personales o identificativos.

La facilidad para la creación de enlaces con otros es una de las principales características disponibles en sitios de SNS, en los que cada usuario dispone de diversas herramientas como son el envío de invitaciones formales, correos electrónicos, mensajería instantánea o disponibilidad de números de teléfono y facilidades para las comunicaciones vocales.

■ Mecanismos de comunicación e interacción entre usuarios:

Los participantes pueden remitir mensajes, comentarios y testimonios, públicos o privados, acerca tanto del perfil del usuario destinatario como de cualquier otro tipo de temática o contenido, incluyendo aquéllos publicados -fotografías, vídeos, etcétera- y los mensajes de otros usuarios. La comunicación cruzada es una actividad fundamental dentro de la red, habilitando canales de comunicación y difusión de mensajes entre sus miembros.

Normalmente, estas anotaciones se muestran en ordenación cronológica inversa por el grado de ocurrencia -la última generada aparece la primera-, y pueden ser persistentes o tener una cierta caducidad (incluso podrían ser borradas instantáneamente por el destinatario si así lo decide). Por tanto, las páginas del usuario contienen los datos e informaciones que éste emplaza, junto a las anotaciones que el resto de los miembros de la red pudieran efectuar, dando pie a una comunicación

cruzada registrada.

- Generación, publicación y consulta de contenidos propios:

El contenido generado por el propio usuario (*User Generated Content*, UGC) permite que éste exprese su creatividad o muestre sus actividades, registrando cualquier tipo de idea o situación imaginable a través de múltiples y variados formatos digitales tales como: los textuales (comentarios y entradas), gráficos (fotografías o ilustraciones), sonoros, vídeos o creaciones multimedia. Este hecho ha conferido un creciente poder al individuo con capacidades de influencia en aspectos sociales, culturales, políticos y económicos, no sólo en su comunidad virtual o en Internet sino también en el mundo físico.

De esta manera es común la publicación -abierta o acotada a un determinado grupo de fotografías o imágenes representativas del individuo, su entorno o situaciones de todo tipo relacionadas con él. También resulta corriente la compartición dentro de la comunidad de sus creaciones textuales y multimedia. En los SNS los usuarios tienen la capacidad de generar contenido (por ejemplo, nuevas entradas en los blogs personales) o más comúnmente, de subirlo a la plataforma para su publicación tras haber sido generado fuera del ámbito de las aplicaciones de los sitios de redes sociales, por ejemplo, vídeos y fotografías.

UGC, al contrario que el contenido editado por expertos profesionales de los medios, no está supeditado a una autoridad con capacidad de condicionarlo según un criterio o propósito corporativo, permitiendo la libertad de expresión del creador. Sin embargo, en caso de publicaciones temáticas este contenido también se considera en muchos casos de menor calidad, credibilidad y fiabilidad que los contenidos profesionales, al no haber superado un alto nivel de supervisión, planteándose también disyuntivas en cuanto a sus derechos intelectuales y de propiedad.

Se estima que el origen de las redes sociales, se remontan al año 1995. Tengamos en cuenta que fue entonces cuando Internet logra convertirse en una herramienta prácticamente masificada. Por aquel entonces, Randy Conrads, quien fue el pionero del servicio, creó el sitio web que llevaba por nombre Classmates, y el cual consistía en una red social que ofrecía la posibilidad de que personas de todo el mundo pudieran recuperar o continuar manteniendo contacto con amigos, compañeros...

A continuación exponemos uno escueto resumen de la evolución de las redes sociales:

- En 1997 la empresa MacroView creó SixDegrees⁵. Fue el primer servicio que combinó todas las características anteriormente descritas, que hasta el momento sólo se habían dado por separado, en un único paquete de servicio. Permitted crear un perfil en la web, agrupar a los contactos directos e intercambiar mensajes con ellos.
- Para 1999 se creó LiveJournal⁶ este fue uno de los primeros servicios de redes sociales en ofrecer blogs y diarios en línea. En el mismo año también se lanza BlackPlanet⁷, exclusivamente diseñada y destinada a afroamericanos, con más de 16.000.000 usuarios a principios de 2008.
- En el 2001 aparece Ryce.com, y nace con ella la primera red social especializada, en la actualidad ésta agrupa por lo menos a 500 mil profesionales.
- En el 2002 nace Friendster⁸, que fue la primera en tener un sistema inteligente capaz de relacionar a los usuarios de la red según sus gustos. También en este año se lanza Fotolog⁹, la primera red social que nace con el fin de intercambiar fotografías.
- En el 2003 se crea Linked in¹⁰, una red social con el fin de propiciar las relaciones empresariales. En ese mismo año nace MySpace¹¹, tal vez la red más grande del momento agrupando a más de 120 millones de usuarios registrados. También en el 2003 nace Hi5¹², que tiene más de 70 millones de cuentas activas. Y MSN Spaces, rebautizado como Windows Live Spaces¹³.
- Flickr¹⁴ nace en el 2004 como una red social que funciona a través del servicio de compartir fotografías. En este año también nace Orkut¹⁵, la red social de Google con gran popularidad en Brasil y La India. Y Facebook¹⁶, la sensación del momento también nace en el mismo año, tiene más de 630 millones de cuentas activas y 50 millones de usuarios diarios.

⁵<http://sixdegrees.com>

⁶<http://www.livejournal.com>

⁷<http://www.blackplanet.com>

⁸<http://www.friendster.com>

⁹<http://www.fotolog.com>

¹⁰<http://www.linkedin.com>

¹¹<http://www.myspace.com>

¹²<http://www.hi5.com>

¹³<http://windowslive.com>

¹⁴<http://www.flickr.com>

¹⁵<http://www.orkut.com>

¹⁶<http://www.facebook.com>

- En el 2005 nace Yahoo 360, la red social de esta empresa que cuenta con servicio de correo, calendario, fotos, eventos, blog y música. También nace Bebo, una red muy básica con el fin de contactar amigos e intercambiar multimedia.
- En el 2006 nace Twitter¹⁷, un microblog, donde no puedes escribir más de 140 caracteres, algo similar a enviar mensajes de texto pero en una PC o laptop.
- Y la última en la cadena fue Xing¹⁸, esta nació en el 2006 con el fin de agrupar ejecutivos permitiéndoles compartir datos, crear agendas conjuntas y construir discusiones en torno a un tema común.

Además de estas redes, cabe destacar otras como son You Tube, Yahoo Answer, Taringa, Matroblog, Neurona, Tuenti, eConozco, Cielo, Networking Activo... Que actualmente han pasado a ser también redes sociales.

La mayor explosión de las redes sociales sucede con el surgimiento de algunos de los sitios más populares que lograron hacer crecer exponencialmente el uso del servicio, comunidades tales como MySpace, Friendster, Facebook, Twitter y Xing, entre otras. Fueron precisamente estos sitios los responsables en lograr que las redes de interacción o círculos de amigos comenzaran a socializarse, con lo que captaron la atención de miles de millones de usuarios de todo el planeta.

Un informe¹⁹ elaborado por la consultora InSites Consulting en el año 2010 devela que en el mundo existen 940 millones de personas registradas a las redes sociales. De acuerdo a Alexa²⁰ (web especializada en la obtención de datos de tráfico de los sitios web) en España las 6 redes más visitadas en el 2011 son Facebook²¹, You Tube²², Windows Live, Twitter, Blogger y Tuenti²³.

2.2.1. Facebook

A día de hoy es la red social más grande del mundo. Creada en febrero de 2004 por Mark Elliot Zuckerberg con la ayuda de Dustin Moskovitz y

¹⁷<http://twitter.com>

¹⁸<http://www.xing.com>

¹⁹<http://www.slideshare.net/stevenvanbelleghe/social-networks-around-the-world-2010>

²⁰<http://www.Alexa.com>

²¹<http://www.facebook.com>

²²<http://www.youtube.com>

²³<http://www.tuenti.com>

Chris Hughes, y con oficinas en Palo Alto, California, inicialmente era una aplicación orientada a ser el lugar de encuentro en la Web de las relaciones en el mundo real de los estudiantes de Harvard. Permitía la conexión en el ciberespacio de los compañeros, donde podían compartir resúmenes, noticias, recordatorios de exámenes, trabajos, etcétera., por lo que su utilización estaba restringida en origen a los estudiantes de dicha universidad bajo invitación a la participación; hasta que, de una manera lenta pero progresiva, los estudiantes empezaron a explorar una nueva faceta de la red dándole un matiz más social, subiendo fotos y datos. Comprobado el éxito generado tras este giro inesperado, Facebook se lanzó dando rienda suelta a esta nueva faceta. En abril de 2004, el equipo de Facebook comenzó a ampliar su público objetivo pasando a estar presente en diversas universidades estadounidenses. En septiembre de dicho año se inició una táctica de expansión entre universidades, colegios mayores e institutos de todo EEUU, Canadá, Reino Unido, Irlanda, México, Puerto Rico, Australia y Nueva Zelanda, cerrando 2005 con presencia en alrededor de 800 universidades, 2.000 colegios mayores y 25.000 institutos. En el año 2006 continuó la expansión frenética de Facebook en India, Alemania e Israel, y ya en 2007 se consolidó en el resto de Europa. En octubre de 2007 Microsoft compró un 1,6 % de Facebook por 240 millones de dólares, como alianza estratégica con propósitos publicitarios y de desarrollo de mercado. Actualmente es la principal red social (a nivel de aplicación web, a nivel general la mayor red social es la formada por los millones de mensajes SMS que se envían las personas con los teléfonos móviles) y ha alcanzado un ámbito mundial.

Por mencionar algunas de las características de Facebook cabría destacar que simplemente añadiendo a un contacto se abre una importante red de conocidos, pues se podrá ver los contactos de éste y, si deseamos, agregarlos. Y podemos proceder del mismo modo con este nuevo contacto. Además, el sistema realiza recomendaciones automáticas de personas que “tal vez conozcas”, basándose en los datos de tu perfil y del de tu posible amigo, logrando contactar con personas que con las que se ha podido perder el contacto. Permite difundir un mensaje de estado en la red, subir fotos y vídeos desde el ordenador o desde el teléfono móvil y compartirlo con tus amigos. Acceder a aplicaciones como juegos, posiblemente desarrollados por terceros ajenos a Facebook, y competir con tu red de contactos. Puedes comentar entradas ('posts') y mandar mensajes privados a otros usuarios.

Resulta llamativo el hecho que sus usuarios por término medio cuenten con un promedio de 195 agregados. Entre las actividades más destacadas que realizan se encuentra el envío de mensajes, conversaciones en los cambios de estado con los contactos, pero gana especial importancia la incorporación a páginas y a grupos, aspecto muy relacionado con técnicas de social media.

En cuanto a conexiones diarias, presenta una media de dos conexiones al día por usuario. Además cuenta casi un millón de páginas de negocios.

El ámbito social que en el que se envuelven los usuarios en Facebook hace propicio mostrar gran cantidad de información personal en los perfiles además de su nombre, su género, su edad, su localización, su trabajo, sus estudios, incluso en algunos casos su situación sentimental, sus preferencias sexuales y políticas. Además un listado de sus gustos musicales, sus películas y libros preferidos, o lugares que les gustan. Los grupos en los que forman parte, etcétera. Con la información que un usuario deposita en el perfil, o el número de personas que se han hecho fan de un producto o se han unido a un grupo de interés en el producto, se puede realizar una inicialización del modelado del usuario (importante para las recomendaciones individuales).

2.2.1.1. Recomendación grupal en Facebook

Los sistemas de recomendación han sido propuestos para aprovechar el potencial de la red social mediante el filtrado de la información y así poder ofrecer recomendaciones que satisfagan a los usuarios. El filtrado colaborativo se cree que es una técnica adecuada que subyace de los sistemas de recomendación dentro de la red social, ya que el filtrado colaborativo recoge los gustos de usuarios similares, y la red social proporciona un entorno de colaboración social.

Los recomendadores de grupo están diseñados para ayudar a aquellos grupos de personas que comparten una actividad común a poder decidirse en situaciones de conflicto. Como se presentó anteriormente, existen multitud de enfoques para hacer recomendaciones a grupos de personas. Trabajos como ARISE (Quijano-Sánchez et al., 2010), (Quijano-Sánchez et al., 2011b), (Quijano-Sánchez et al., 2011a) están basados en tres características principales: personalidad, confianza social (trust) y memoria de recomendaciones anteriores. De esta manera, tratan de simular de la manera más realista posible el proceso de argumentación seguido por los grupos de personas a la hora de decidir una actividad conjunta. Ahora, nosotros, en este proyecto, tratamos de dar un escenario real, a través de la red social Facebook, a las teorías expuestas por ARISE para hacer recomendaciones grupales.

Vendría bien mencionar dos interesantes estudios. Primero haremos referencia a *Collabio* (Bernstein et al., 2009). Es una aplicación de Facebook, lanzada por investigadores de Microsoft, diseñada para explorar los metadatos y la lógica relacionada con el etiquetado. Anima a los usuarios a adivinar las etiquetas que otros amigos han usado para describir a un individuo. Dichos etiquetados se agregan a una nube de palabras, al principio ocultando

las letras de la palabra con imágenes. Los puntos se otorgan de acuerdo al número de amigos que adivinan dichas etiquetas coincidiendo con las dejadas por otros (cuando se adivina la etiqueta se muestra en la nube) o proporcionando nuevas etiquetas, que definen a la persona que se añaden a la nube. El tamaño de la palabra es más grande cuantas más personas crean que la etiqueta es la adecuada para la persona; y los puntos que obtienen las personas al descubrir las etiquetas será mayor cuanto mayor sea la palabra).

En el estudio las personas encuestadas afirman que las etiquetas de las nubes les definen correctamente, y que existen etiquetas que si bien ellos creen que no los definen, coinciden en que son apreciaciones que los demás suelen tener sobre ellos. La aplicación mantiene marcadores de puntuación a nivel local y global, y el estudio muestra que la motivación social de los participantes evita los problemas de las inconsistencias, falsedades o malos usos en las valoraciones de los usuarios. Resulta verdaderamente atractiva la forma de agregación de las etiquetas, ideas u opiniones de un grupo de personas en forma de nube, y se podría usar para mostrar una agregación de recomendaciones. Dar puntos por valorar recomendaciones, o por ser flexible para llegar a un acuerdo final también podría ser interesante. *Collabio* también se basa en motivaciones sociales como la reciprocidad, los usuarios son avisados cuando los amigos les etiquetan y le anima a etiquetar en respuesta.

Por su parte, en el estudio *Group Recommendation System for Facebook* (Baatarjav et al., 2008), se realizan recomendaciones de grupos a personas en función del perfil de la persona. Es decir en función del perfil individual de una persona se le recomienda un grupo, teniendo en cuenta el perfil del grupo recomendado. Dicho perfil de grupo se obtiene a partir de los perfiles de las personas que son miembros de dicho grupo, para ello se realiza un *clustering* (grupos social con rasgos e intereses comunes) jerárquico de dichos perfiles donde se eliminan las características de los usuarios que no son relevantes para la mayoría del grupo. Posteriormente para recomendar un grupo a un individuo utilizan un árbol de decisión para a partir de las características del perfil del individuo recomendarle un grupo. Realizaron el estudio en la red de Facebook de la Universidad del Norte de Texas de Estados Unidos, a partir de los datos de 1580 usuarios y 17 grupos de intereses comunes existentes en dicha red. En sus conclusiones muestran que el sistema tiene una precisión en sus recomendaciones del 64 %, y del 73 % (aumento del 9 %) después de eliminar el ruido de las características de usuarios no relevantes en un grupo.

Resulta de utilidad ver cómo es posible realizar un modelado de un grupo a partir de los perfiles de sus usuarios, sin embargo se puede mejorar la forma en que se realiza el modelado del grupo con taxonomías (Gómez-Gauchía et al., 2006) y/o con los métodos de agregación usados en el estudio (Recio-

García et al., 2009), teniendo en cuenta la personalidad de los miembros del grupo en situaciones de conflicto para poder determinar qué recomendaciones grupales se adecuarían mejor a la interacción de un grupo de individuos, así como el temperamento de cada uno de los integrantes del grupo.

Según los autores de (Álvarez Arrieta et al., 2010) podemos confirmar que la interacción social necesaria en un sistema de recomendación grupal, se traslada de la aplicación web comercial, a un medio donde dicha interacción social entre los miembros de un grupo se realiza de forma habitual y mejor, la red social. Este enfoque aporta beneficios tanto a los usuarios (que se ahorran el tener que perder tiempo introduciendo información de sus preferencias en cada negocio web), para el sistema de recomendación (ahorro del tiempo y esfuerzo de desarrollo necesarios para que el sistema gestione las preferencias de los usuarios individuales, y proporcione formas de comunicación grupales para las decisiones de grupo), para los negocios (las recomendaciones pueden llegar a un número de personas muchísimo mayor), y para la red social al ofrecer un nuevo y mejor servicio social a sus usuarios.

2.2.1.2. Aplicaciones. Test de personalidad

Existen multitud de aplicaciones desarrolladas para las redes sociales. Muchas están muy extendidas y son muy populares entre los usuarios, llegando a sobrepasar los 400 millones de usuarios (*Farm Ville*) 2.6. Éstas abarcan diferentes índoles: juegos, aplicaciones afectivas, tests de personalidad y parentesco, etcétera. Pero por norma general todas cumplen ciertos patrones que las hermanan:

- Un sistema de puntuaciones y diferentes niveles de juego. Con el fin de mantener el interés del usuario.
- Poder notificar las acciones del juego en el muro para fomentar la interacción social.
- Poder invitar a los amigos a la aplicación a través de la red social.
- Posibilitar la interacción de los usuarios a través del chat, posts o mensajes privados.
- Por regla general, suelen ser aplicaciones muy básicas y de fácil manejo.

Nótese, que de existir, deben ser pocas las aplicaciones que tengan en cuenta la personalidad del usuario. Para nuestro proyecto como ya mencionamos, el enfoque se basaba en 3 pilares: personalidad, confianza social (trust) y memoria de recomendaciones anteriores. Para la personalidad



Figura 2.6: Imagen del juego FarmVille

creemos necesario realizar un cuestionario previo el cual nos permita modelar el temperamento del usuario. Indagando en Facebook encontramos que existen un gran número de aplicaciones que realizan un cuestionario previo de 10 a 15 preguntas al usuario, tras las cuales les muestra qué personaje de determinada serie de televisión, película, personaje histórico, animal o color *son*. Les explica las virtudes del personaje emparentado. Dicho resultado se revela en su página principal de Facebook, de forma que todos sus amigos puedan verlo. Se suele mostrar una imagen del personaje y la descripción del mismo. Un ejemplo es, *¿Qué personaje de Perdidos eres?* que permite al usuario saber a qué personaje de la serie de televisión Perdidos (*Lost*) se parece.

Para obtener la personalidad del usuario y sus preferencias, los usuarios deben contestar dos test diferentes. El primero es el test de personalidad, donde los usuarios tienen que elegir entre una serie de personajes con los que se sienten identificados (módulo de personalidad), como se muestra en la Figura 3.13. Una vez han contestado el test de personalidad, los usuarios tienen que votar una colección de películas (al menos 20 películas), donde sugieren cuáles son sus preferencias personales (este es el test Web que usa el módulo de estimación individual), como se muestra en la Figura 3.16.

Estas aplicaciones tienen mucho éxito porque resulta bastante ameno a la par de curioso realizar los tests y descubrir con qué personaje te relaciona. Que todos tus amigos puedan ver a quién te pareces fomenta la interacción social. El inconveniente, como decíamos, es que no se utiliza esta información para realizar un modelado de la personalidad del usuario, pero a nosotros nos

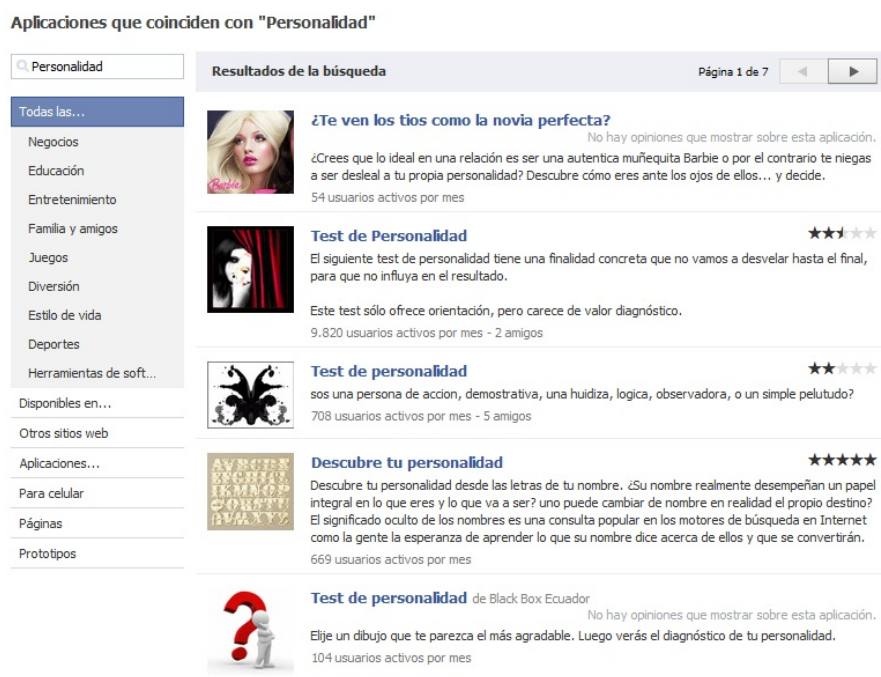


Figura 2.7: Aplicaciones de test de personalidad

resulta de gran inspiración pues parece original poder reflejar el resultado con una imagen de una persona con aspecto formal para representar que el usuario tiene un comportamiento en el que predomina el componente racional. Para fomentar la diversión y la interacción entre los usuarios mostraremos la imagen de un personaje famoso que tenga el mismo temperamento que el usuario.

La aplicación *Personalidad* (2.7) ofrece una gran variedad de tests de personalidad. Como por ejemplo: *¿Qué sabor de helado eres?* *¿A qué personaje de una serie de dibujos animados te pareces?*. El test suele constar de unas 10 a 15 preguntas. El orden de las preguntas del test es aleatorio para evitar la monotonía en el caso que se vuelva a realizar. Se ofrece una imagen y una breve descripción de la personalidad resultante.

La originalidad de esta aplicación es que permite buscar y contactar a personas que han realizado el mismo test y que han obtenido resultados similares a los nuestros. También podemos ver el grado de similitud de personalidad que tenemos con el resto de usuarios de la aplicación.

El inconveniente que presentan vuelve a ser que no hacen ninguna mención a la personalidad, como se explica en la Sección 3.2.3.1 del Capítulo 3.

Figura 2.8: *Events*

2.2.1.3. Creación de eventos

Como ya sabemos, en Facebook existen diversas formas de interactuar con otros usuarios, ya sea, por ejemplo, por el chat, los grupos, las páginas o los eventos que podemos crear disponiendo de un perfil en la Red Social. Todos ellos ayudan a crear un espacio para intercambiar información, intereses, opiniones...

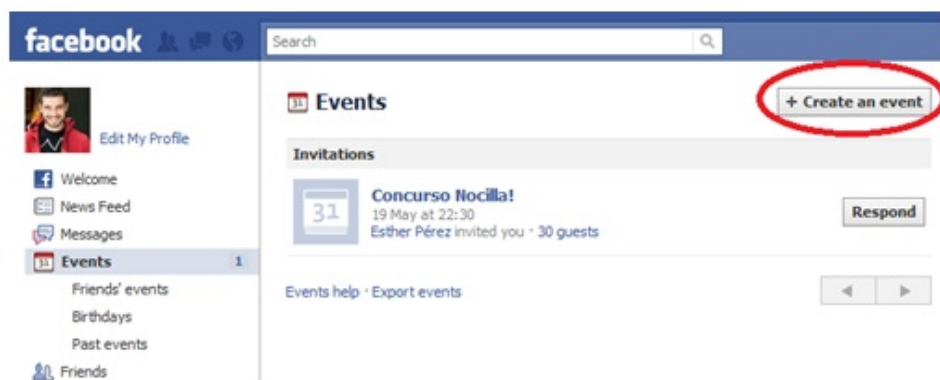
A continuación explicaremos cómo se crean y cómo funcionan los eventos en Facebook, pues nuestro proyecto tratará de imitar sus características para aplicarlas en los eventos de nuestra aplicación.

El éxito cosechado por la herramienta “Crear un evento” en Facebook se debe a que puesto que la Red Social nos permite reunir en un sólo lugar a todos nuestros amigos, resulta lógico utilizarla a la hora de organizar un encuentro, reunión o cualquier celebración, ya que la mayoría de los invitados pertenecen a ella.

Para ello está la función *Events* (Eventos), donde podemos planear todos los detalles para cualquier tipo de encuentro social en persona, incluyendo quienes asistirán; dándoles la oportunidad de confirmar su asistencia e informarse acerca de cómo llegar, entre otras cosas.

A diferencia de otras funciones, *Events* viene por defecto en todas las cuentas de Facebook y no es necesario añadirla como Aplicación.

Para crear un evento deberá, primero acceder a su perfil (cuenta) de

Figura 2.9: *create event*

Facebook y pulsar sobre *Events*, para luego hacer clic sobre *Create an event* (Crear un evento, Figura 2.9).

El proceso de Crear un Evento consta de la especificación del evento, la política de privacidad que se le quiere dar y la selección de los invitados.

En la especificación del evento se indicará la información básica que permitirá determinar el día de inicio del evento (*When?*), cuándo finaliza (*End time?*), en qué consiste el evento (*What are you planning?*), y dónde tendrá lugar (*Where?*). Opcionalmente se puede añadir la dirección exacta, Street. Y se deberá especificar la localidad donde tendrá lugar (*City/Town*).

Conforme a la privacidad nos dan la opción de aceptar que cualquiera pueda ver y responder al evento (evento público), seleccionando la casilla *Anyone can view and RSVP (public event)*; y la opción de mostrar la lista de invitados en la página del evento, marcando la casilla *Show the guest list on the Event Page*(Figura 2.10).

Por último, y casi es lo más importante, es necesario seleccionar los contactos que estarán invitados (*Who's invited?*), pinchando en *Select guest*(1), Seleccionar amigos. Se puede ir uno por uno o indicar que toda la lista está invitada. Además es posible agregar el correo electrónico de otras personas que no están en Facebook, pero que también se desea que asistan, añadiendo si se desea un pequeño mensaje personal(Figura 2.11).

Para finalizar, seleccionará *Create event* (2) y listo. Ahora los invitados podrán confirmar su asistencia, indicar que no irán o simplemente decir que no están seguros. De esta manera el creador del evento podrá estimar cuanta gente asistirá al mismo.

facebook

Search

Home Profile Find friends Account

Create an event

When? 6/26/2011 10:00

End time 6/26/2011 14:30

What are you planning? Cine en vez de examen de PUG

Where? Kinépolis Madrid

Street Edgar Neville Ciudad de la Imagen s/n.

City/Town Pozuelo de Alarcón, 28223

More info? Y después, comemos y besamos :D

Who's invited?

Select guests (1)

☒ Anyone can view and RSVP (public event)

☒ Show the guest list on the Event Page

Create event (2)

Figura 2.10: *Guest list event*

Invite friends

Search by name Search all friends

<input type="checkbox"/> Anabel	<input type="checkbox"/> Claudia	<input type="checkbox"/> Elihu
<input checked="" type="checkbox"/> Enrique	<input type="checkbox"/> Esther	<input type="checkbox"/> Esther
<input type="checkbox"/> Francisco Calvo	<input type="checkbox"/> Ilona	<input checked="" type="checkbox"/> Javier
<input type="checkbox"/> Jenny	<input type="checkbox"/> Lara	<input type="checkbox"/> Leti
<input type="checkbox"/> Lucía	<input type="checkbox"/> Marta	<input type="checkbox"/> Marta
<input type="checkbox"/> Mimi	<input type="checkbox"/> Raquel Helena	<input type="checkbox"/> Web Amaia

Invite by email address: (use commas to separate email addresses)

melondieselsandiasinplomo@noescoña.com

Add a personal message

Save and close

Figura 2.11: *Guest list*

Figura 2.12: *Event*

La página que verán los invitados lucirá de la siguiente manera, teniendo la oportunidad de editarla cada vez que lo deseen (Figura 2.12).

Opcionalmente podrá añadir un poco de información al evento. Además tendrás la posibilidad de que la página del evento pueda tener Muro, vídeos, fotos, enlaces...

2.3. Tecnologías usadas

En esta sección describiremos las tecnologías utilizadas en nuestro proyecto, todas de ellas exceptuando SQL, que previamente habíamos utilizado para la gestión de bases de datos, eran desconocidas para nosotros antes comenzar el desarrollo de *HappyMovie* lo que requirió una labor importante de investigación y documentación, especialmente para el uso de JavaBridge 2.3.4 ya que es una tecnología que no está ampliamente extendida y de la que no se dispone gran cantidad de información.

2.3.1. PHP

2.3.1.1. Descripción

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente para la interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en

la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas *Qt* o *GTK+*.

PHP es un acrónimo recursivo que significa *PHP Hypertext Pre-processor* (inicialmente *PHP Tools*, o, *Personal Home Page Tools*). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de *PHP* es producida ahora por The *PHP Group* y sirve como el estándar de facto para *PHP* al no haber una especificación formal. Publicado bajo la *PHP License*, la *Free Software Foundation* considera esta licencia como software libre.

Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. El lenguaje *PHP* se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores, el número de sitios en *PHP* ha compartido algo de su preponderante sitio con otros nuevos lenguajes no tan poderosos desde agosto de 2005. Es también el módulo *Apache*²⁴ más popular entre las computadoras que utilizan *Apache* como servidor web.

2.3.1.2. Aplicación

PHP es la columna vertebral de toda aplicación de *Facebook*, gracias a esta tecnología y a la SDK que *Facebook* nos proporciona podemos manejar de manera relativamente sencilla la API de *Facebook*. Toda aplicación de Facebook requiere de una página índice(*Index.php*) que será la primera página a la que *Facebook* llamara una vez ejecutemos nuestra aplicación, en esta página nos encargaremos del manejo del inicio de sesión del usuario y de otros aspectos fundamentales para la aplicación.

2.3.2. JavaScript

2.3.2.1. Descripción

JavaScript es un lenguaje de programación interpretado, dialecto del estándar *ECMAScript*. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (*client-side*), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, aunque existe una forma de *JavaScript* del lado del servidor (*Server-side JavaScript* o *SSJS*). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones

²⁴<http://www.apache.org>

de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y *JavaScript* no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código *JavaScript* integrado en las páginas web. Para interactuar con una página web se provee al lenguaje *JavaScript* de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente sin acceso a funciones del servidor. *JavaScript* se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML.

2.3.2.2. Aplicación

A parte de la SDK de *PHP* nombrada anteriormente, *Facebook* nos proporciona también una SDK de *JavaScript*, dicha SDK se está convirtiendo cada día más, en una de las más utilizadas en detrimento de la SDK de *PHP*, ya que facilita mucho el manejo de la API de *Facebook*.

En nuestra aplicación hemos usado esta tecnología para manejar el redimensionado de todas las páginas de la aplicación así como para dar funcionalidad a los distintos botones de la misma.

Una de las grandes ventajas de esta SDK es que se puede cargar de manera asíncrona, esto quiere decir que no tiene que esperar a que la página este completamente cargada, agilizando así la velocidad de carga de la página.

2.3.3. AJAX

2.3.3.1. Descripción

Ajax, acrónimo de Asynchronous *JavaScript* And XML (*JavaScript* asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las apli-

caciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se requieren al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. *JavaScript* es el lenguaje interpretado (*scripting language*) en el que normalmente se efectúan las funciones de llamada de *Ajax* mientras que el acceso a los datos se realiza mediante *XMLHttpRequest*, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dado que está basado en estándares abiertos como *JavaScript* y Document Object Model (DOM).

2.3.3.2. Aplicación

Esta tecnología ha sido usada para incluir un sistema de votación por estrellas para las películas propuestas por el recomendador grupal, dicho sistema requiere de una tecnología asíncrona, que permita realizar cambios en la página una vez has realizado las votaciones, sin necesidad de recargar la página.

2.3.4. JavaBridge

2.3.4.1. Descripción

PHP/JavaBridge es una implementación de streaming, con un protocolo en red basado en XML, el cual se puede conectar a un motor script nativo, por ejemplo *PHP*, *Scheme* or *Python*, con una máquina virtual de Java. Es casi 50 veces más rápida que una local RPC via SOAP, y requiere menos recursos en el lado del servidor web. Es más rápida y más fiable que la comunicación directa vía la Interfaz Nativa de Java, y no requiere componentes adicionales para invocar a procedimientos de Java desde *PHP* o procedimientos *PHP* hacia Java. El clustering de soporte J2EE y el balance de descarga de Apache también están incluidos dentro de las aplicación a las que se puede aplicar *JavaBridge* así como los Scripts de *PHP* usando JSP, JSF u otros frameworks.

2.3.4.2. Aplicación

JavaBridge es una de las tecnologías más importantes utilizadas en nuestro proyecto, esta tecnología nos permite invocar a clases java dentro de *PHP*,

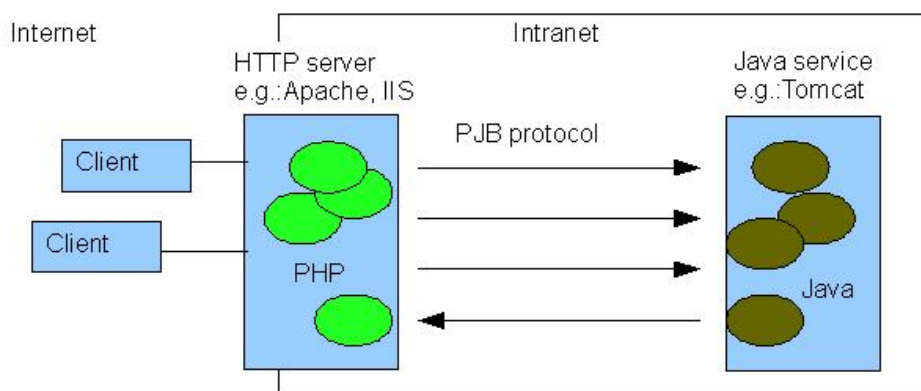


Figura 2.13: Funcionamiento de Javabridge en *HappyMovie*

lo cual es fundamental ya que todos los sistemas de recomendación que empleamos están escritos en Java.

Empleando esta tecnología podemos crear clases java desde *PHP* de manera sencilla y llamar a métodos de dichas clases con variables *PHP* como parámetros.

2.3.5. SQL

2.3.5.1. Descripción

El lenguaje de consulta estructurado o SQL (por sus siglas en inglés structured query language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en éstas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar -de una forma sencilla- información de interés de una base de datos, así como también hacer cambios sobre ella.

Los orígenes del SQL están ligados a las de las bases de datos relacionales. En 1970 E. F. Codd propone el modelo relacional y asociado a éste un sub-lenguaje de acceso a los datos basado en el cálculo de predicados. Basándose en estas ideas, los laboratorios de IBM definen el lenguaje SEQUEL (Structured English QUery Language) que más tarde sería ampliamente implementado por el sistema de gestión de bases de datos (SGBD) experimental System R, desarrollado en 1977 también por IBM. Sin embargo, fue Oracle quien lo introdujo por primera vez en 1979 en un programa comercial.

El SEQUEL terminaría siendo el predecesor de SQL, siendo éste una versión evolucionada del primero. El SQL pasa a ser el lenguaje por excelencia de los diversos sistemas de gestión de bases de datos relacionales surgidos en los años siguientes y es por fin estandarizado en 1986 por el ANSI, dando lugar a la primera versión estándar de este lenguaje, el “SQL-86” o “SQL1”.

El SQL es un lenguaje de acceso a bases de datos que explota la flexibilidad y potencia de los sistemas relacionales permitiendo gran variedad de operaciones en éstos últimos.

Es un lenguaje declarativo de “alto nivel” o “de no procedimiento”, que gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros, y no a registros individuales, permite una alta productividad en codificación y la orientación a objetos. De esta forma una sola sentencia puede equivaler a uno o más programas que se utilizarían en un lenguaje de bajo nivel orientado a registros.

2.3.5.2. Aplicación

El lenguaje SQL ha sido usado para la gestión de la base de datos de nuestro servidor, tanto desde dentro de la propia aplicación mediante consultas a esta, como desde fuera de la aplicación utilizando programas gestores de bases de datos como son SQL-Front y Heidi SQL, Figuras 2.14 y 2.15. Estos programas nos han permitido verificar que tanto las consultas como el almacenamiento de información en la base de datos se realizaban correctamente, además de permitirnos modificar manualmente algunos parámetros de la base de datos para la depuración de errores.

MySQL-Front - lara_server - s05ad7ee_laradb.película (Explorador de datos)

Archivo Editar Buscar Ver Favoritos Base de Datos Extras Configuración Ayuda

Explorador de objetos Explorador de datos Editor SQL

ID	Título	Año	Genero	Duración	Calificación	Nacionalidad	Url
1	127 horas	2011	Drama	01:34:00	Mayores de 12 años	Reino Unido, EE.UU.	http://www.guiadelo
2	Águila Roja: La peli	2011	Acción	02:02:00	Mayores de 12 años	España	http://www.guiadelo
3	Bebés	2011	Documental	01:19:00	Todos los públicos	Francia	http://www.guiadelo
4	Bienvenidos al Sur	2011	Drama	02:00:00	Todos los públicos	Italia	http://www.guiadelo
5	Caperucita Roja (LA	2011	Fantástica	01:40:00	Mayores de 12 años	EE.UU.	http://www.guiadelo
6	Código Fuente	2011	Acción	01:33:00	Mayores de 12 años	EE.UU.	http://www.guiadelo
7	Country Strong	2011	Drama	01:57:00	Mayores de 7 años	EE.UU.	http://www.guiadelo
8	Crebinsky	2011	Comedia	01:30:00	Todos los públicos	España	http://www.guiadelo
9	El amor y otras cosas	2011	Drama	01:59:00	Mayores de 7 años	EE.UU.	http://www.guiadelo
10	El discurso del rey	2010	Drama	01:58:00	Todos los públicos	Reino Unido	http://www.guiadelo
11	El mundo según Ban	2011	Drama	02:12:00	Mayores de 7 años	Canadá, Italia	http://www.guiadelo
12	En un mundo mejor	2011	Drama	01:59:00	Mayores de 7 años	Dinamarca	http://www.guiadelo
13	Fast and Furious 5	2011	Acción	02:10:00	Mayores de 12 años	EE.UU.	http://www.guiadelo
14	Furia ciega 3D	2011	Acción	01:44:00	Mayores de 16 años	EE.UU.	http://www.guiadelo
15	Gnomeo y Julieta	2011	Animación	01:24:00	Todos los públicos	EE.UU.	http://www.guiadelo
16	Hop	2011	Familiar	02:00:00	Todos los públicos	EE.UU.	http://www.guiadelo
17	La legión del Águila	2011	Acción	01:54:00	Mayores de 12 años	Reino Unido, EE.UU.	http://www.guiadelo
18	No tengas miedo (20	2011	Drama	01:30:00	Mayores de 16 años	España	http://www.guiadelo
19	Perdona pero quiero	2011	Comedia	01:46:00	Todos los públicos	Italia	http://www.guiadelo
20	Rio	2011	Animación	01:36:00	Todos los públicos	EE.UU.	http://www.guiadelo
21	Scream 4	2011	Terror	02:00:00	Mayores de 16 años	EE.UU.	http://www.guiadelo
22	Sin límites (2011)	2011	Misterio	01:45:00	Mayores de 12 años	EE.UU.	http://www.guiadelo
23	The company men	2011	Drama	01:44:00	Mayores de 7 años	EE.UU.	http://www.guiadelo
24	Thor	2011	Acción	02:10:00	Mayores de 7 años	EE.UU.	http://www.guiadelo
25	Una dulce mentira	2011	Drama	01:45:00	Todos los públicos	Francia	http://www.guiadelo
26	Invasión a la Tierra	2011	Acción	01:56:00	Mayores de 12 años	EE.UU.	http://www.guiadelo
27	Soy el número cuatr	2011	Acción	01:49:00	Mayores de 12 años	EE.UU.	http://www.guiadelo
28	Tormenta 4 : Lethal ci	2011	Comedia	01:30:00	Mayores de 12 años	España	http://www.guiadelo
29	Winnie the Pooh	2011	Animación	01:06:00	Todos los públicos	EE.UU.	http://www.guiadelo
30	Ahora los padres sor	2010	Comedia	01:40:00	Todos los públicos	EE.UU.	http://www.guiadelo
31	Amor y otras drogas	2011	Comedia	01:52:00	Mayores de 7 años	EE.UU.	http://www.guiadelo
32	Cartas a Dios	2011	Drama	01:45:00	Todos los públicos	Bélgica, Francia	http://www.guiadelo
33	Cane negro	2011	Drama	01:48:00	Mayores de 12 años	EE.UU.	http://www.guiadelo
34	El rito (2011)	2011	Thriller	01:54:00	Mayores de 16 años	EE.UU.	http://www.guiadelo
35	Esta abuela es mi pa	2011	Comedia	01:47:00	Todos los públicos	EE.UU.	http://www.guiadelo
36	Happy Thank You M	2011	Comedia	01:40:00	Todos los públicos	EE.UU.	http://www.guiadelo
37	Incendies	2011	Drama	02:10:00	Mayores de 12 años	Canadá, Francia	http://www.guiadelo

Figura 2.14: MySQL Front

supergala/s05ad7ee_laradb - HeidiSQL 6.0.0.3603

File Edit Search Tools Import Export Help

Host: localhost Database: s05ad7ee_laradb Query

Name	Rows	Size	Created	Updated	Engine	Comment	Type
actores	50	10,9 KB	2011-03-27 18:15:11	2011-03-27 18:26:31	MyISAM		Table
actuan	613	64,0 KB	2011-05-05 12:39:29		InnoDB	Actore...	Table
attendance	3	4,3 KB	2011-03-27 18:15:12	2011-06-07 18:30:10	MyISAM		Table
directores	50	4,5 KB	2011-03-27 18:15:13	2011-03-27 18:26:31	MyISAM		Table
dirigen	105	32,0 KB	2011-05-05 12:39:29		InnoDB	Direct...	Table
evento	1	2,1 KB	2011-06-03 11:09:37	2011-06-03 11:09:40	MyISAM		Table
gruposabiertos	0	4,0 KB	2011-06-23 18:59:03	2011-06-23 18:59:03	MyISAM		Table
guionistas	50	5,0 KB	2011-03-27 18:15:14	2011-03-27 18:26:31	MyISAM		Table
invitacion	1	2,1 KB	2011-03-27 18:15:14	2011-06-07 18:30:10	MyISAM		Table
movies_info	50	7,4 KB	2011-03-27 18:15:14	2011-03-27 18:26:31	MyISAM		Table
película	117	80,0 KB	2011-05-05 12:39:28		InnoDB	Inform...	Table
pelprovincia	1,207	48,0 KB	2011-05-05 12:39:29		InnoDB	pelcul...	Table
provincia	50	16,0 KB	2011-05-05 12:39:28		InnoDB	Nombr...	Table
ratedmovies	121	11,1 KB	2011-03-27 18:15:14	2011-06-07 19:22:58	MyISAM		Table
scores	6	2,2 KB	2011-03-27 18:15:14	2011-03-27 18:15:21	MyISAM		Table
trust	3	2,1 KB	2011-03-27 18:15:16	2011-03-27 18:15:21	MyISAM		Table
users	12	2,5 KB	2011-03-27 18:15:16	2011-06-07 18:30:10	MyISAM		Table
usuariosgrupo	4	4,1 KB	2011-03-27 18:15:16	2011-06-06 12:58:53	MyISAM		Table
votes	0	1,0 KB	2011-03-27 18:15:16	2011-03-27 18:15:16	MyISAM		Table

Figura 2.15: Heidi SQL

Capítulo 3

El prototipo

3.1. Teoría. Modelo teórico

Los objetivos de este proyecto se centran en el estudio de los factores sociales en los sistemas de recomendación, para así reflejar las interacciones que realmente se producen entre los usuarios cuando se desarrolla un proceso de recomendación o negociación al realiza una actividad social. Además, se ha incluido un segundo factor que valora la influencia de los distintos perfiles de personalidad de cada usuario involucrado en el proceso(Recio-García et al., 2009).

Con este trabajo, se pretende mejorar las técnicas existentes que normalmente se emplean a la hora de recomendar distintos tipos de productos a grupos (Jameson y Smyth, 2007). Una de las principales tendencias cuando se realizan recomendaciones es usar agregación de *ratings* para individuos. Sin embargo, la estrategia de agregación ha sido criticada por algunos autores ya que los *ratings* tienden a combinarse siempre del mismo modo sin tener en cuenta las relaciones entre los miembros del propio grupo. Para solucionar este problema nosotros proponemos en primer lugar, incluir las interacciones, lo que significa que realizamos las recomendaciones usando información personal sobre el comportamiento en situaciones de conflicto para cada uno de los miembros del grupo; y segundo, incluir información sobre cómo interactuarán los mismos.

El proceso de recomendación tendrá en cuenta la personalidad de cada individuo, es decir, atendiendo a la forma en la que reaccionará cuando el sistema le recomiende una propuesta diferente a la que él quería veremos si el usuario es abierto de mente, si colabora en la toma de la decisión, etcétera. También evaluará la confianza existente entre los componentes del grupo. Por ejemplo, si el cliente se fía de la opinión de otro miembro, si existe habitualmente coincidencias de opiniones con algún otro usuario, etcétera. De

esta manera conseguimos reflejar, de la forma más realista posible, los procesos reales de recomendación y negociación sociales llevados a cabo por las personas.

3.2. Desarrollo del prototipo

3.2.1. Nuestra aplicación: *HappyMovie*

HappyMovie es una instanciación de la arquitectura genérica ARISE (Quijano-Sánchez et al., 2011c) usando *Facebook*. Nuestro objetivo es trasladar los típicos sistemas locales (almacenado en un único ordenador) a una nueva clase de sistemas Web donde las relaciones sociales se tengan en cuenta en el proceso de recomendación para grupos. Con este tipo de aplicaciones somos capaces de ofrecer recomendaciones a grupos para todas las personas conectadas a la red social, además podemos obtener mucha información de ésta sin tener que molestar a nuestros usuarios con muchos cuestionarios, también nos proporciona mucho *feedback*¹ lo que nos permite mejorar nuestra metodología.

Para usar nuestra aplicación, los usuarios simplemente tienen que abrir su cuenta en Facebook y buscar *HappyMovie* en la sección de aplicaciones. Explicaremos cómo usar la aplicación a través de un ejemplo. Supongamos que tenemos un grupo dado conectado en la red social como vemos en la Figura 3.1. Los pasos necesarios para obtener una recomendación de una película para un grupo con *HappyMovie* se explican a continuación:

- **Crear un perfil de usuario en la aplicación:** Antes de que ningún usuario pueda acceder a los resultados de la recomendación de películas, debemos crear su “perfil de recomendación” individual pues es necesario para nuestro método de recomendación. Este perfil está basado en tres aspectos diferentes: personalidad, preferencias individuales y confianza con los otros usuarios.

Los usuarios que no hayan rellenado estos dos test, no podrán acceder a ninguna otra funcionalidad de la aplicación (invitar amigos, aceptar invitaciones a eventos, crear eventos, acceder a ningún evento o recomendación, etcétera). Finalmente, para obtener el último factor –la confianza– la aplicación lee la información almacenada en el perfil personal de *Facebook* de cada usuario. Y se calcula la confianza que cada

¹El *feedback* (retroalimentación) es el proceso de compartir observaciones, preocupaciones y sugerencias, con la intención de recabar información, a nivel individual o colectivo, para intentar mejorar el funcionamiento de una organización o de cualquier grupo formado por seres humanos.

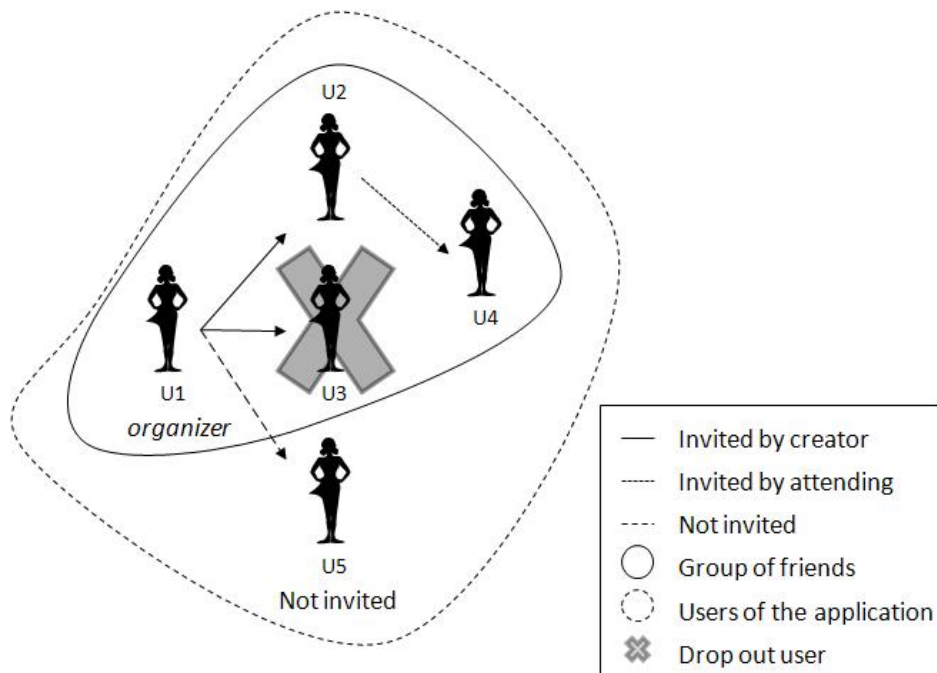


Figura 3.1: Amigos conectados a través de Facebook

usuario tiene con todos los demás usuarios que se ha unido al evento hasta ese momento.

- **Crear una actividad:** El usuario organizador, u_1 decide organizar una actividad e inicia la aplicación para crear un nuevo evento “*vayamos al cine!*”, la Figura 3.2 muestra cómo se presenta esta opción en *HappyMovie*. Para crear un evento, los organizadores deben establecer los datos de configuración deseados: lugar, fecha, nombre del evento, etcétera. Una vez se ha creado el evento el organizador puede invitar a cualquiera de sus amigos de Facebook a él. Cualquier usuario que asista al evento puede ver la fecha de realización, la fecha límite para apuntarse, el lugar y el nombre del creador, así como una propuesta de tres películas.

Estas tres opciones de películas son las mejores que el recomendador grupal ha encontrado para los usuarios que en ese momento asistirán al evento. Por ejemplo, cuando el usuario u_1 inicia la aplicación por primera vez, tiene un rol de *organizador*. Como *organizador* primeramente invita a sus amigos al evento. Digamos que invita a u_2 y u_3 . Una vez que el *organizador* realiza la configuración inicial del evento continuará usando la aplicación pero con un rol de *usuario común*.

facebook

Search

Lets go to the cinema!

Hello everyone! I'm creating this event so that we can go all together to the cinema, I hope you all join!

Name of the Event

When?

MM DD YYYY

Where?

Madrid

Last day to join the event

MM DD YYYY

Checkboxes

☐ Show the guest list of the event

☐ Anyone can see and join the event(public event)

Create Event Back

Figura 3.2: Formulario crear Evento

Cuando los usuarios participan en un evento como *usuario común* también pueden invitar a cualquiera de sus amigos Facebook al evento y si desean pueden borrarse del evento en cualquier momento (antes de la fecha, deadline). Por ejemplo, u_2 acepta la invitación de ir al evento y más tarde invita a su amiga en Facebook u_4 . Por otro lado, u_3 acepta inicialmente la invitación y se une al evento, pero más adelante decide no ir, así que se borra de él.

- **Recomendación:** Una vez que la aplicación obtiene los tres factores que identifican a cada usuario que se une al evento (personalidad, preferencias individuales y confianza con los otros usuarios) proporciona una recomendación grupal usando uno de nuestros métodos explicados en la Sección 3.2.3.6.

- **Recibir una recomendación:** Cuando se crea el evento busca en la cartelera de cine actual en la ciudad indicada y muestra una lista de 3 películas, que representan las 3 mejores películas que el recomendador ha encontrado en la cartelera para los usuarios que se han unido al evento hasta ese momento, esto se muestra en la Figura 3.3. Esta lista se modifica automáticamente cada vez que un usuario se une al evento o se borra de él. Este proceso continua siempre hasta el día en el que el *organizador* ha señalado como fecha límite (deadline date). En nuestro ejemplo, inicialmente se realiza una recomendación para los usuarios u_1 , u_2 y u_3 cuando se unen inicialmente al evento. Luego, cuando u_3 se borra del evento se realiza una nueva recomendación para u_1 y u_2 . Finalmente cuando u_4 , que fue invitado por u_2 , se une al evento una nueva colección de 3 películas aparece en el evento para los usuarios u_1 , u_2 and u_4 .

Cuando llega el día final para la realización del evento se puede ver la lista final de películas recomendadas. Desde el mismo día del deadline los usuarios pueden votar cada una de las películas propuestas por medio de una encuesta que se visualiza en ese momento. Sólo se podrá realizar la votación una única vez, momento en el que la encuesta ya no volverá a estar disponible. Este proceso nos permite decidir qué película se verá finalmente, que será la película que obtenga la votación más alta en media y más importante nos proporciona el *feedback* necesario para evaluar la calidad de nuestra recomendación.

3.2.2. Fases de desarrollo de la aplicación *HappyMovie*

3.2.2.1. Fase de investigación y documentación

A la hora de desarrollar nuestra aplicación una de las fases más importantes y laboriosas fue la búsqueda de información sobre como desarrollar una aplicación en *Facebook*. El principal problema fue que la información que se puede encontrar en la Web esta muy desorganizada, a pesar de los últimos esfuerzos que *Facebook* esta haciendo por centralizar en su Web toda la información, todavía resulta bastante complicado encontrar algunas informaciones vitales para el desarrollo de aplicaciones dentro de *Facebook*.

A esto hay que unir el hecho de que *Facebook* esta realizando cambios muy importantes en el modo de desarrollar aplicaciones en su plataforma. Esta plataforma que *Facebook* lanzo en 2007 como una nueva herramienta para desarrolladores que permitía integrar con la popular aplicación *Facebook*, ha cambiado sustancialmente a lo largo del tiempo, tratando de convertirse en

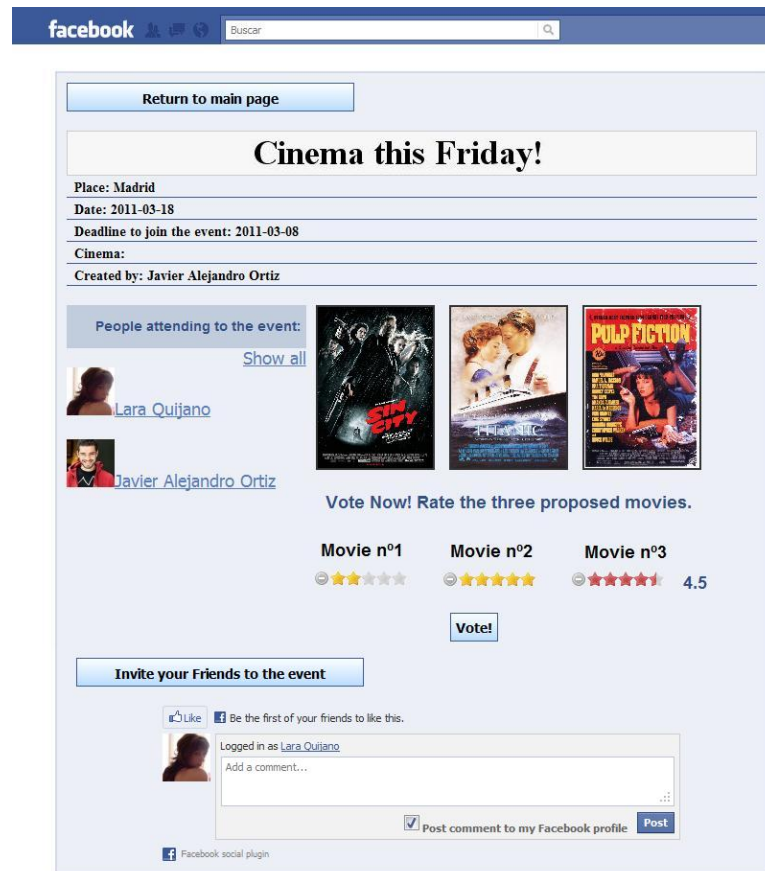
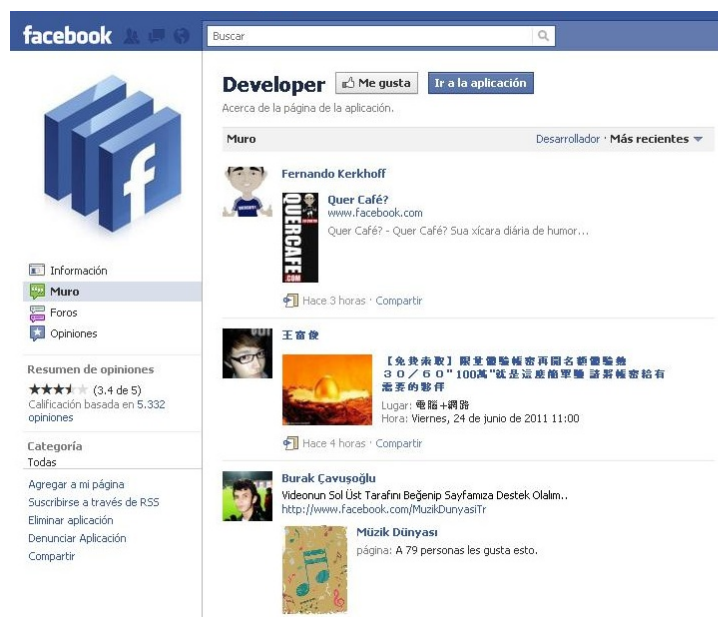


Figura 3.3: Evento con recomendación

una herramienta mas potente y sencilla que permitiese a los desarrolladores crear nuevas aplicaciones.

Tanto es así, que hoy en día esta plataforma se encuentra en pleno proceso cambio desde que *Facebook* anuncio el lanzamiento de su nueva API, conocida como "*Graph API*", gracias a esta API cualquier Web puede "socializarse" de manera relativamente sencilla y los procesos con usuarios se vuelven más sencillos, lo que permite a más Webs adaptarse a *Facebook* e integrarlo totalmente en sus sistemas.

El principal problema que presenta este cambio es que la nueva API todavía no esta completamente terminada y seguimos teniendo que utilizar la que se ha denominado como *Old REST API* para poder realizar algunas tareas.

Figura 3.4: Aplicación *Developer*

3.2.2.2. Creación y configuración de la aplicación

Para poder empezar a trabajar en nuestro proyecto lo primero que era necesario hacer era crear una nueva aplicación en Facebook. Para ello es necesario tener una cuenta en Facebook y agregar la aplicación denominada “Developers”, Figura 3.4 . Una vez que hemos agregado esta aplicación tenemos que pulsar el botón “*set up new application*”, Figura 3.5, y configurarla, para ello debemos introducir la url de nuestro servidor, así como seleccionar el tipo de aplicación que queremos desarrollar.

Facebook nos permite elegir entre FBML e IFRAME, nosotros seleccionamos esta última ya Facebook está dejando de dar soporte a las aplicaciones de tipo FBML obligando así a todos los desarrolladores a crear nuevas aplicaciones de tipo IFRAME que permiten integrar Webs y aplicaciones ya creadas con gran facilidad, Figura 3.6.

Una de las opciones más interesantes que Facebook nos da a la hora de configurar nuestra aplicación es la de seleccionar la opción avanzada “*remove deprecated auth methods*”, que siguiendo el mapa de ruta de Facebook nos impide utilizar métodos que dejaran de tener soporte próximamente. Figura 3.7.

En un principio alojamos la aplicación en un servidor Web ajeno a la facultad, en dicho servidor disponíamos de soporte *PHP* y *MySQL* (2.3),

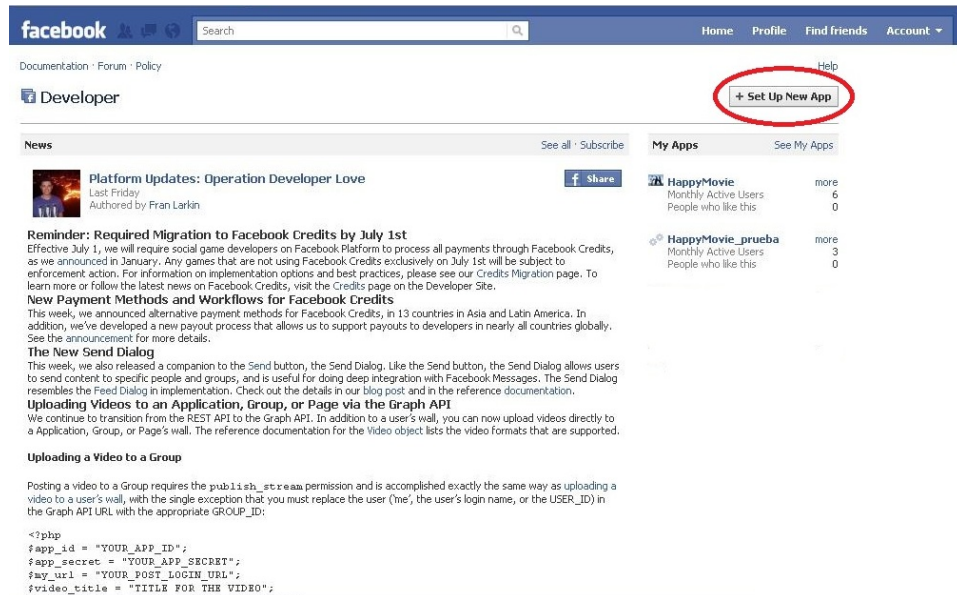


Figura 3.5: Nueva aplicación

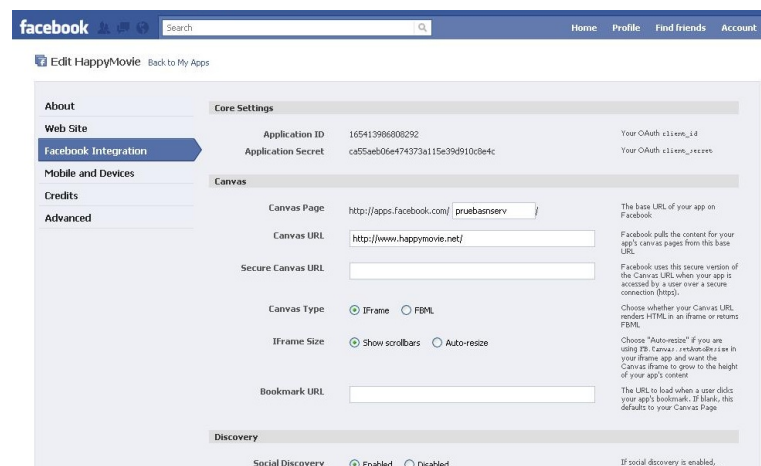


Figura 3.6: Integración en Facebook

Edit HappyMovie Back to My Apps

About

Web Site

Facebook Integration

Mobile and Devices

Credits

Advanced

Authentication

Deauthorize Callback

Sandbox Mode ☐ Enable ☒ Disable

Migrations

Remove Deprecated Auth Methods ☒ Enabled ☐ Disabled

Stream post URL security ☐ Enabled ☒ Disabled

OAuth 2.0 for Canvas ☒ Enabled ☐ Disabled

Timezone-less events ☐ Enabled ☒ Disabled

Upgrade to Requests 2.0 ☐ Enabled ☒ Disabled

Security

Server Whitelist

Update Settings IP Whitelist

Update Notification Email

Advertising

Advertising Accounts

Facebook pings this URL when a user deauthorizes your app.

If enabled, only the developers of your app will be able to sign in to your app.

With this Enabled, you can't use deprecated methods on the deprecation roadmap.

Links in stream posts must go to the application's canvas or connect URL. Includes a 'signed_request' parameter for canvas apps.

Facebook events do not support time zones. Accordingly, event start_time and end_time are now returned from the graph API without a time zone.

Bookmark count represents number of all outstanding requests.

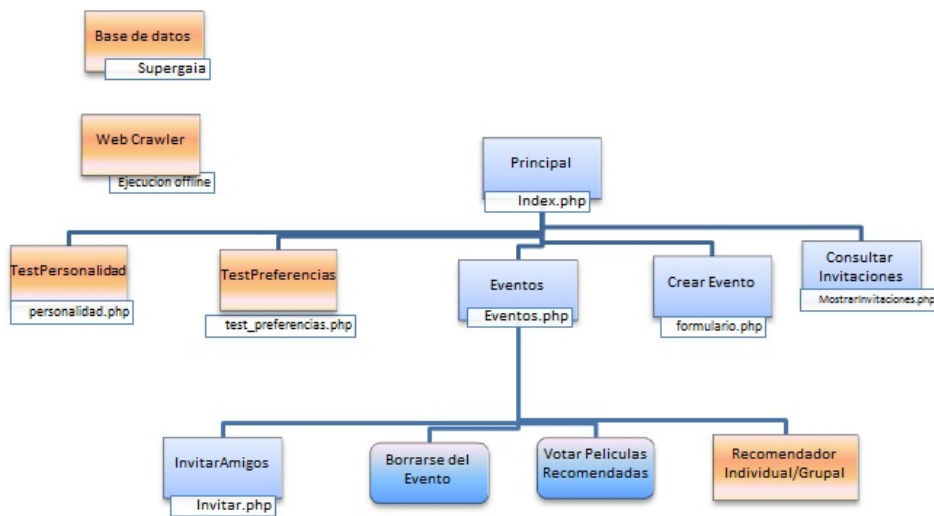
If specified, Facebook will block requests from your app unless they come from an IP in this comma-separated list.

If specified, Facebook will block changes to app settings unless they are made from an IP in this comma-separated list.

Facebook sends any updates made to app settings to this email address.

Comma-separated list of Facebook advertising accounts associated with your app. Enter the email address.

Figura 3.7: Opciones avanzadas

Figura 3.8: Organigrama *HappyMovie*

dos tecnologías que como ya se han mencionado antes son fundamentales en nuestra aplicación. Sin embargo, como se explicara más adelante, debido a las exigencias de nuestro proyecto más adelante sería necesario un cambio de servidor.

3.2.2.3. Fase de desarrollo e implementación

La implementación de nuestra aplicación está basada en una estructura de archivos *PHP* que escriben dinámicamente el contenido de HTML. El estilo de las páginas está definido usando hojas de estilo CSS. En la Figura 3.8 se presenta la estructura básica de nuestra aplicación, remarcándose en color **naranja** los cinco módulos fundamentales de la misma (Base de datos, Web Crawler, TestPersonalidad, TestPreferencias y Recomendador Individual/Grupal).

A continuación se describen tanto el contenido de las carpetas de nuestra aplicación como algunos de los PHP más importantes de la misma:

- **Root:**

- *Config.php:*

Se llama en todos los archivos *PHP* mediante la función *require_once* y se encarga de la configuración de los parámetros fundamentales de la aplicación, como son la api key (clave que *Facebook* asigna a

nuestra aplicación) y la *appid* (numero único que distingue nuestra aplicación del resto de aplicaciones de *Facebook*). También se encarga del control de inicio de sesión de los usuarios, comprobando que cuando un usuario acceda a la aplicación haya iniciado sesión en *Facebook*.

- *Index.php*:

Es la página principal de la aplicación, en ella se consultan tanto los eventos a los que asistirá el usuario como las invitaciones que ha recibido para unirse a un evento. A través de esta página se da acceso al resto de la aplicación.

- *Action_lib.php*:

En esta página están agrupadas todas las funciones de consulta a la base de datos de la aplicación, que permiten manejar toda la información de los usuarios, algunos ejemplos son: *registra_usuario* (*user_id*, *personality*, *nombre*), *existe_usuario* (*user_id*), *listar_eventos* (*user_id*), etc. En esta página también se incluyen todas las funciones necesarias para el cálculo del trust.

- *Funciones.php*:

Es la página que contiene todo lo referido a los recomendadores: funciones para el cálculo de grupos, para la generación de archivos XML, parseo de información.

- *Style.css*:

Hoja de estilos css que se utilizan en la aplicación.

- **Lib:**

Aquí almacenamos todas las librerías de la aplicación, así como las SDKs de Facebook.

- **Formulario:**

Guarda el formulario de creación de eventos.

- **TestPersonalidad:**

Contiene una página *index.php*, que se encarga de cargar dentro de Facebook la aplicación flash que se encargara de calcular la personalidad de cada usuario. En el archivo *scoresDB.php* se calculara la personalidad del usuario y se almacenara en la base de datos.

- **TestPreferencias:**

Esta carpeta contiene tanto la página PHP que se emplea para mostrar una colección de películas que el usuario deberá votar como la que se encarga de almacenar las votaciones en la base de datos.

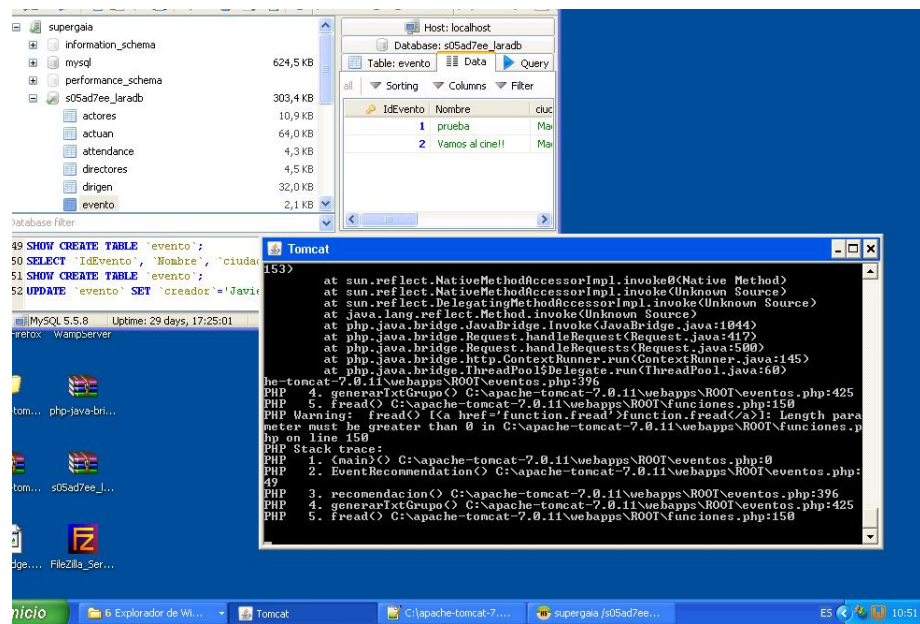


Figura 3.9: Máquina virtual

Por último es importante mencionar que, como hemos comentado anteriormente, fue necesario un cambio de servidor a mitad de la fase de desarrollo de la aplicación. El problema radicaba en que nuestra aplicación necesita invocar a clases Java dentro de PHP y nuestro primer servidor no permitía la integración de *JavaBridge*(2.3).

Actualmente nuestra aplicación se encuentra alojada dentro de una de las máquinas virtuales del servidor de la Facultad en Informática de la Universidad Complutense de Madrid². En esta máquina virtual configuramos un servidor *Apache* así como el módulo *JavaBridge*, que se encuentra almacenado dentro de la carpeta *webapps*. Figura 3.9. Después de realizar estos cambios y debido a las restricciones de *Supergaia* fue necesario crear un dominio *HappyMovie.net* y configurar de nuevo nuestra aplicación para que accediese al servidor a través de este nuevo dominio.

²supergaia.fdi.ucm.es:2250

3.2.2.4. Fase de pruebas

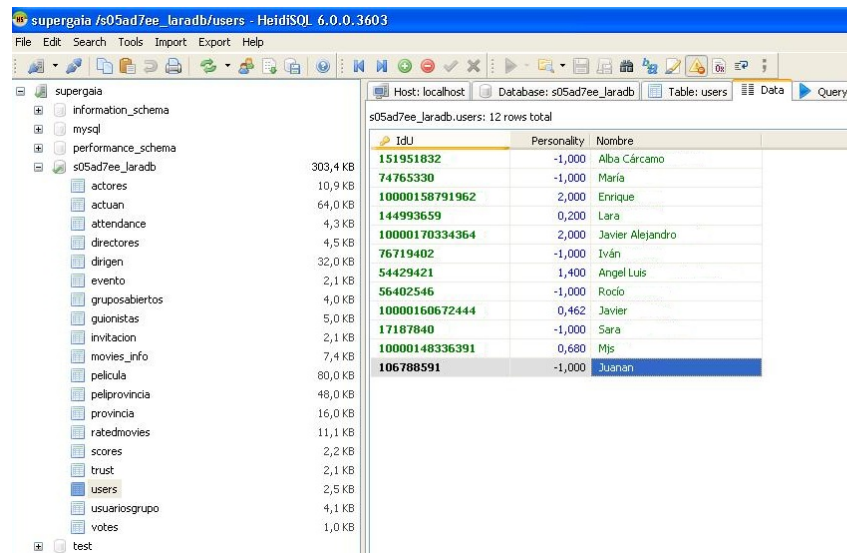
Para la realización de la fase de pruebas contamos con la colaboración de un grupo de amigos que tenían cuenta en *Facebook* y a los que se les envió una invitación para usar la aplicación. La primera etapa de la fase de pruebas consistió en que los usuarios probaran el funcionamiento básico de la aplicación:

- Dar de alta a un usuario que no existiese ya en la base de datos y almacenarlo en la base de datos (tabla users): Figura 3.10 .
- Crear un evento y comprobar que se cumplen los plazos de fecha de *deadline*: Figura 3.11 .
- Listar correctamente todos los eventos a un usuario ha indicado que asistirá así como toda la lista de invitados a ese evento junto con los links a sus perfiles de *Facebook*.
- Mostrar correctamente las invitaciones a los usuarios, tanto en Facebook como una vez dentro de la aplicación dando la posibilidad de rechazar o aceptar la invitación así como de realizar los test de personalidad y preferencias en caso de que no se hubiesen completado todavía.
- Comprobar que la personalidad se calcula correctamente y se almacena en la base de datos una vez realizado el test de personalidad.
- Comprobar que las votaciones del sistema de estrellas del test de preferencias se almacenan correctamente así como que la barra de progreso se actualiza con cada votación.

Como ya veremos más adelante, las técnicas de recomendación grupal usando factores de personalidad y confianza mejoran significativamente los resultados obtenidos por otros sistemas de recomendación grupal, sin es necesario comprobar que este fundamento teórico tiene unos resultados que se ajustan a los obtenidos en un caso de uso real.

Así pues era necesaria una segunda etapa de pruebas que además de garantizar el funcionamiento básico de la aplicación demostrase que las recomendaciones recibidas por los usuarios eran satisfactorias.

Esta segunda etapa que aún se está llevando a cabo consiste en el desarrollo de un experimento real guiado en el que pediremos a los usuarios que creen eventos y formen grupos siguiendo unas pautas fijadas y a los que posteriormente se les realizaran una serie de cuestiones para evaluar el grado de satisfacción con los resultados de la aplicación.

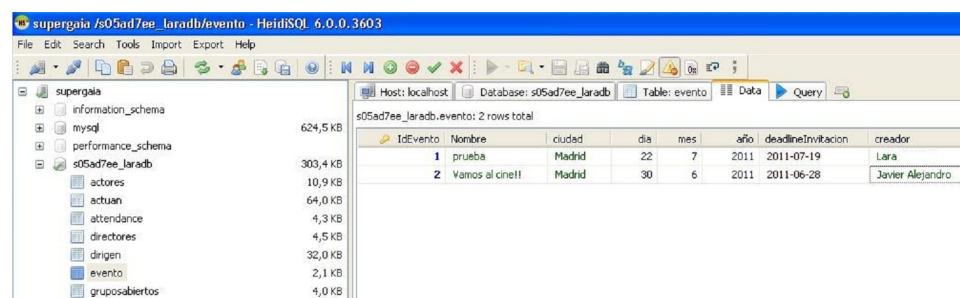


supergaia /s05ad7ee_laradb/users - HeidiSQL 6.0.0.3603

Host: localhost Database: s05ad7ee_laradb Table: users Data Query

s05ad7ee_laradb.users: 12 rows total

IdU	Personality	Nombre
151951832	-1,000	Alba Cárcamo
74765330	-1,000	María
10000158791962	2,000	Enrique
144993659	0,200	Lara
10000170334364	2,000	Javier Alejandro
76719402	-1,000	Iván
54429421	1,400	Angel Luis
56402546	-1,000	Rocio
10000160672444	0,462	Javier
17187840	-1,000	Sara
10000148336391	0,680	Mjs
106788591	-1,000	Juanan

Figura 3.10: Bases de datos *Usuarios*


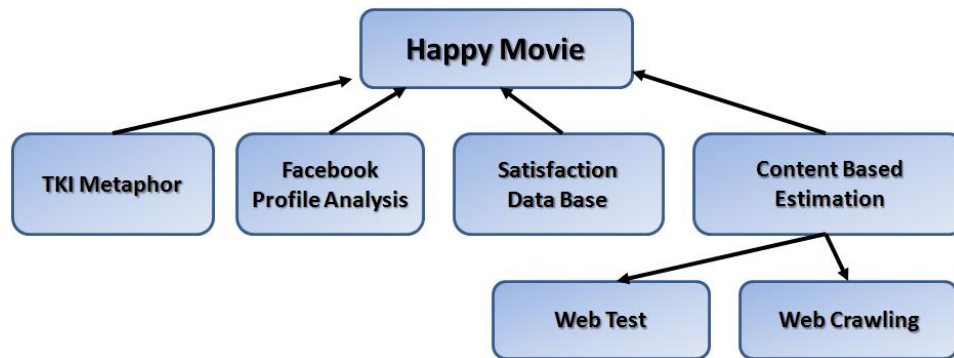
supergaia /s05ad7ee_laradb/evento - HeidiSQL 6.0.0.3603

Host: localhost Database: s05ad7ee_laradb Table: evento Data Query

s05ad7ee_laradb.evento: 2 rows total

IdEvento	Nombre	ciudad	dia	mes	año	deadlineInvitacion	creador
1	prueba	Madrid	22	7	2011	2011-07-19	Lara
2	Vamos al cine!!	Madrid	30	6	2011	2011-06-28	Javier Alejandro

Figura 3.11: Bases de datos *Evento*

Figura 3.12: Arquitectura *HappyMovie*

3.2.3. Arquitectura. Módulos *HappyMovie*

Los factores sociales (contagio emocional, confianza, personalidad...) son una pieza fundamental a la hora de recomendar productos para grupos de personas. Sin embargo, la complejidad de manejar estos factores radica en que son muy difíciles de estimar, ya que para su obtención, las personas deben estar dispuestas a realizar encuestas o cuestionarios que a menudo son bastante molestos para los usuarios. Sin embargo, hoy en día la red colaborativa nos proporciona una herramienta muy útil para realizar esta tarea de una manera amena: las redes sociales. Las redes sociales permiten a los usuarios interactuar y desarrollar sus relaciones sociales a través de la Web. A este fenómeno se le conoce como Web 2.0. Creemos que este lugar es el más adecuado para inferir los elementos sociales nombrados anteriormente. Por ejemplo, podemos estimar la intensidad del vínculo entre dos usuarios midiendo el número de mensajes intercambiados, o conociendo el número de amigos comunes. Este análisis nos permite explotar la información disponible en la Web para realizar recomendaciones a grupos mucho más completas y más precisas.

Nuestra arquitectura está basada en ARISE (*Architecture for Recommendations Including Social Elements*), que se aplica a aquellos sistemas recomendadores de grupos que tienen en cuenta la composición de la personalidad del grupo y las conexiones sociales entre los componentes del grupo. Además incluye un dispositivo de memoria de las recomendaciones ya realizadas que permite dar más peso a las preferencias de los usuarios insatisfechos y conseguir así un sistema con un cierto grado de justicia para el grupo. A lo largo de este capítulo presentamos *HappyMovie*, una aplicación de *Facebook* para realizar recomendaciones de películas a grupos de usuarios.

Modalidad TKI	Autoritarismo		Cooperacionismo	
	Alto	Bajo	Alto	Bajo
Competitiva	0.375	-0.075	-0.15	0
Colaborativa	0.375	-0.075	0.375	-0.075
Complaciente	0	0	0	0
Evasiva	-0.375	0.075	-0.375	0.075
Comprometida	-0.15	0	0.375	-0.075

Tabla 3.1: Coeficientes usados para calcular el CMW.

3.2.3.1. Módulo Personalidad: Metáfora TKI

Existen diferentes enfoques que nos permiten obtener los distintos roles que se observan en las personas cuando interactúan en un proceso de toma de decisión. Para nuestro módulo de personalidad hemos desarrollado una aplicación en *flash* que utilizamos para evaluar la personalidad de cada usuario. Hemos utilizado una *metáfora de películas* para obtener la personalidad de cada usuario en situaciones de conflicto. Esta metáfora interactiva consiste en mostrar dos personajes de películas con personalidades opuestas para cada una de las cinco categorías de personalidad posibles (competitivo, complaciente, evasivo, colaborativo y comprometido). Un personaje representa unas características esenciales de la categoría, mientras que el otro representa las contrarias. El mecanismo que debe seguir el usuario consiste en elegir, moviendo una flecha, el personaje de los que se ofrece por categoría con el que se identifique en mayor medida. Una vez completado el proceso de la metáfora TKI se obtiene un porcentaje para cada categoría. A partir de esta información podemos reducir de los cinco tipos de personalidad a dos: *Autoritarismo* y *Cooperacionismo*.

- **Autoritarismo.** Penaliza negativamente las diferencias entre las selecciones preferidas de otros miembros.
- **Cooperacionismo.** Recompensa las diferencias con las selecciones preferidas por otros miembros.

Para ello utilizamos los coeficientes de la Tabla 3.1.

$$CMW_i = \frac{1 + \text{Autoritarismo}(u_i) - \text{Cooperacionismo}(u_i)}{2} \quad (3.1)$$

Las “situaciones conflictivas” son aquellas en las que los intereses de dos personas parecen ser incompatibles. En dichas situaciones, como acabamos de comentar, podemos describir el comportamiento de un individuo a lo largo de dos dimensiones básicas:

- Autoritarismo. El grado de predisposición que tiene una persona de satisfacer sus propios intereses.
- Cooperacionismo. El nivel de predisposición que tiene una persona de satisfacer los intereses de otros.

Estas dos dimensiones básicas de comportamiento definen cinco diferentes modos de responder en situaciones de conflicto:

- Competitiva
- Colaborativa
- Evasiva
- Complaciente
- Comprometida

Cada individuo posee estos tipos de respuestas en un determinado grado. Para obtener dicho grado utilizamos el test TKI, que nos devuelve un determinado porcentaje para cada tipo de personalidad por usuario. El test de personalidad TKI, propone 30 situaciones donde el usuario tiene que decidir cómo reaccionaría. Por ejemplo en la pregunta número 7 los usuarios deben decidir con qué estaría más de acuerdo, con "*He procurado posponer la resolución del problema hasta haber tenido algún tiempo para pensar en él*" o con "*He procurado imponer mi decisión*". Como ya hemos apuntado, estas preguntas resultan tediosas y no siempre claras. Por eso hemos creado la metáfora interactiva, esto hará nuestra aplicación más atractiva y amena.

Para realizar la metáfora TKI hemos empleado los siguientes personajes para cada una de las cinco categorías:

- Competitivo:
 1. *Personaje representativo*: Es alguien que pone sus intereses por encima de todo, sin importarle el coste que le suponga a otras personas. Es un tipo de personalidad orientada al poder donde la persona usa cualquier poder del que pueda disponer para conseguir su objetivo: su habilidad discutiendo, su cargo, o su dinero. Es alguien que lucha por sus derechos y por aquello que piensa que es correcto o simplemente intenta ganar. Nuestro ejemplo es *Michael Corleone* de *El Padrino*.
 2. *Personaje opuesto*: Es alguien que nunca se aprovecha de su situación por muy favorable que sea si eso supone alguna contrariedad a otros. Aun pudiendo, ya sea porque tenga la oportunidad o el poder necesario, esta persona hará lo que sea justo incluso si esa

solución no es la más adecuada para él. Nunca impone su opinión y siempre escucha a todo el mundo. Nuestro ejemplo es *Harry Potter* de la película homónima.

■ Complaciente:

1. *Personaje representativo*: Es alguien que ignora sus intereses con tal de satisfacer los de otra persona. En este modo existe un elemento de sacrificio propio. Toma forma de generosidad desinteresada o caridad, obedeciendo lo que diga otra persona aunque personalmente preferiría no hacerlo, o ceder ante el punto de vista de otra persona. Nuestro ejemplo es *Sam Gamgee* de *El Señor de los Anillos*.
2. *Personaje opuesto*: Es alguien que nunca se sacrifica por el equipo. No es ni obediente, ni indulgente o generoso. Siempre se mantiene firme a su opinión. Nuestro ejemplo es el *Doctor House* de la serie de televisión *House*.

■ Evasiva:

1. *Personaje representativo*: Es alguien que no se enfrenta a conflictos. El modo evasivo tomará siempre la forma de la diplomacia haciéndose siempre a un lado cuando el conflicto se produzca, posponiéndolo hasta que la situación sea más adecuada o simplemente retirándose de dicha situación. Nuestro ejemplo es *Jack Sparrow* de *Piratas del Caribe*.
2. *Personaje opuesto*: Es alguien que se enfrenta a todo tipo de situaciones a las que se ve expuesto. Nunca piensa que exista una situación imposible que no se pueda resolver, nunca se rinde hasta que todo tiene un final deseado. Nuestro ejemplo es *John McClane* de *Jungla de Cristal*.

■ Colaborativa:

1. *Personaje representativo*: Es alguien que intenta trabajar con todos para resolver una situación que satisfaga plenamente a todo el mundo. Significa que investiga hasta el fondo todas las posibilidades y sugiere las necesidades prioritarias que tienen todos los individuos. Colaborativo es cuando dos personas exploran los motivos del desacuerdo estudiando todas las posibilidades y las necesidades individuales y finalmente encuentran una solución creativa que resuelva el conflicto. Nuestro ejemplo es *Woody* de *Toy Story*.
2. *Personaje opuesto*: Es alguien que nunca trabaja en equipo. Es individualista, arrogante en cierta medida, nunca escucha al resto

del grupo porque considera que su solución es la mejor. Nuestro ejemplo es *Mark Lenders de Oliver y Benji*.

■ Comprometida:

1. *Personaje representativo*: Es alguien que propone llegar a un acuerdo entre las diferentes opiniones donde todo el mundo haga concesiones. Nunca se rinde o le da la espalda a cualquier conflicto hasta que este se ha resuelto. Pone todo de su parte para intentar resolverlo. Incluso si fuera más fácil abandonar, o las cosas se pusieran difíciles intentaría resolver el problema a cualquier coste. Nuestro ejemplo es *William Wallace de BraveHeart*.
2. *Personaje opuesto*: Es alguien que no hace concesiones. No está comprometido a resolver ningún tipo de situación y si llega a un punto complicado no dudará en abandonar. Lucha por sus propios objetivos y no le gusta trabajar en equipo. Nuestro ejemplo es *Lobezno de X-Men*.

Nuestra interfaz del test de personalidad muestra una página como la que se puede ver en la Figura 3.13. Para que el usuario pueda responder el test, debe pulsar sobre el botón *Start*. Pasará a la primera modalidad TKI donde aparecen las dos primeras fotos de personajes con personalidades opuestas. El individuo deberá desplazar la flecha indicando con mayor o menor medida su grado de identificación con respecto a los personajes. La flecha tiene diez posiciones que representan el porcentaje que el usuario tiene de cada categoría. El personaje que representa la categoría sería un diez en la flecha (en la aplicación podemos ver una foto del personaje y una descripción de él) y el personaje opuesto representaría un cero en la flecha (también podemos ver la información referente a este personaje, en el lado opuesto a la derecha). Una vez hecho esto, deberá pulsar el botón *Next* y repetir el proceso con todas las modalidades. Una vez finalizado el test, la aplicación dispone de la suficiente información para “medir” o “calcular” su personalidad.

Este módulo tiene como objetivo obtener los rasgos característicos de la personalidad del usuario en función a cinco modalidades TKI mencionadas en el capítulo 3, en la figura 3.1, con los pesos mostrados también en la tabla.

Para comenzar a responder al test pulsamos en *Start*, pasando a la primera modalidad TKI donde aparecen la foto de un personaje y su personalidad, siendo éstas totalmente opuestas. En este punto del test nuestro quehacer es según nos sintamos indentificados de mayor o menor medida desplazar la flecha, donde la flecha indica el grado de identificación con respecto a la personalidad del personaje. Una vez seleccionado dicho grado, se pulsa *Next* pasando así por todas las modalidades, cuando se ha finalizado el test, para

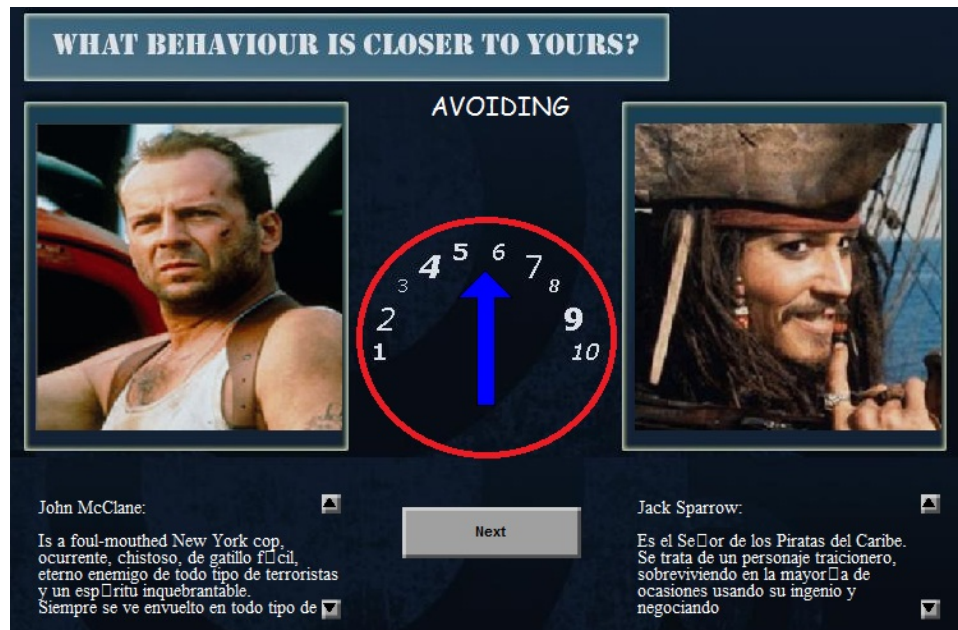


Figura 3.13: Página test de personalidad

calcular la personalidad se aplica la Fórmula 3.1.

Fundamento teórico del módulo de personalidad

La mayoría de los trabajos anteriores sobre recomendaciones grupales, consideran las preferencias de cada miembro del grupo con el mismo grado de importancia e intentan satisfacer las predilecciones de cada uno, véase (McCarthy y Anagnost, 1998; McCarthy, 2002) o (Lieberman et al., 1999). Sin embargo, hay que considerar que no todos los grupos son iguales: tienen diferentes características (como el tamaño, por ejemplo) o pueden estar formados por personas con preferencias similares o antagónicas (grupo homogéneo o heterogéneo respectivamente). Cuando nos enfrentamos a una situación en la que las inquietudes de las personas son incompatibles surge una situación conflictiva. Los sistemas recomendadores para grupos existentes generalmente consiguen resolver este tipo de conflictos, intentando maximizar las preferencias del mayor número de miembros del grupo. Pero, es sabido que la satisfacción general del grupo no se corresponde siempre con la agregación de las satisfacciones de sus miembros, pues cada persona tiene diferentes expectativas y comportamientos en situaciones de conflicto, por lo que éstas se deben tener presentes para las recomendaciones.

Test de Personalidad

Pregunta 1

- ☒ En ocasiones he dejado a los otros la responsabilidad de encontrar la solución a los problemas.
- ☐ En vez de negociar sobre los aspectos en los que hemos estado de acuerdo, he preferido aclarar bien los aspectos en los que si hemos estado de acuerdo

Pregunta 2

- ☒ Dependiendo de la situación, algunas veces he luchado por los resultados y en otras, más me ha importado mantener la buena relación.
- ☐ He intentado tomar en cuenta tanto mis inquietudes como también las de la otra parte.

Pregunta 3

- ☒ Generalmente he estado muy decidido a lograr mis objetivos.
- ☐ A veces he intentado calmar los sentimientos del otro y mantener nuestra relación.

Figura 3.14: Ejemplo de preguntas realizadas en el test TKI

Para el desarrollo del módulo de personalidad de la aplicación nos hemos basado en resultados obtenidos en estudios anteriores (Recio-García et al., 2009),(Quijano-Sánchez et al., 2010),(Quijano-Sánchez, 2010) que utilizaban el test TKI (Thomas y Kilmann, 1974), que está diseñado para la medición del comportamiento humano en situaciones conflictivas (véase Figura 3.14). El TKI es un instrumento puntero en los estudios de determinación de conflictos ya que describe cómo actúa una persona ante situaciones encontradas por medio de dos dimensiones básicas: autoritarismo y cooperacionismo. Estas dos dimensiones de comportamiento nos permiten definir los distintos tipos de personalidad existentes en las situaciones de conflicto (competitivo, complaciente, evasivo, colaborativo y comprometido).

El método denominado como *Personality Aware Recommendation to Groups*, tiene en cuenta los diversos tipos de personalidades. El TKI se encarga de construir un perfil de usuario a través de 30 preguntas de respuesta múltiple. El test proporciona puntuaciones para los cinco tipos de personalidad que representan la tendencia del individuo. Las puntuaciones se normalizan para obtener percentiles usando una muestra de 8000 personas. Comentar, que el TKI es usado por asesores de Recursos Humanos y Desarrollo Organizativo debido a su utilidad para aprender sobre los conflictos que se pueden producir en dinámicas personales y de grupo, para saber cómo afrontarlos.

El principal problema de los estudios nombrados anteriormente era que los usuarios encontraban el test tedioso, largo y, en ciertas ocasiones, con-

fuso en algunas de las preguntas. Para solucionar este problema y hacer la aplicación más dinámica y entretenida se ha optado por usar la *metáfora de películas* como método alternativo para obtener la personalidad de los usuarios en situaciones de conflicto (Sección 3.2.3.1). Se ha comprobado en (Quijano-Sánchez et al., 2011b) que es posible reemplazar el test original con este nuevo porque los resultados obtenidos en ambos tests resultan equivalentes.

3.2.3.2. Confianza: Análisis de los perfiles de usuarios en *Facebook*

El módulo de la Confianza es el módulo que obtiene los mayores beneficios al tener la aplicación sobre una red social, en nuestro caso, *Facebook*. Anteriormente, con una aplicación local, la tarea de obtener los datos necesarios para calcular el *trust* entre usuarios era muy aburrida. Ahora, somos capaces de calcular el *trust* entre usuarios extrayendo la información específica que se encuentra alojada en cada uno de los perfiles de los usuarios dentro de la red social.

Para poder acceder a la información que necesitamos de nuestros usuarios, estos han de autorizarnos a leer la información almacenada en sus perfiles de *Facebook* antes de empezar a utilizar la aplicación. Este proceso es igual para todas las aplicaciones de *Facebook*, cuando los usuarios acceden a cualquier aplicación, éstas les solicitan una serie de permisos para poder extraer datos de su perfil. En nuestro caso *HappyMovie* solicitará los siguientes permisos:

- Permisos para el acceso a las fotos del usuario: user photo video tags, friends photo video tags, user photos: friends photos: nos permiten acceder tanto a las fotos subidas por el usuario como a las fotos en las que ha sido etiquetado.
- Permisos para el acceso a la lista de amigos del usuario: read friend _lists: nos permite leer la lista de amigos de un usuario.
- Permisos para el acceso al muro del usuario: read stream: nos permite leer el muro de un usuario. publish stream: nos permite publicar en el muro de un usuario.

Nosotros en *HappyMovie* analizamos los siguientes factores:

- Amigos en común.
- Fotos en común.
- Intereses comunes:

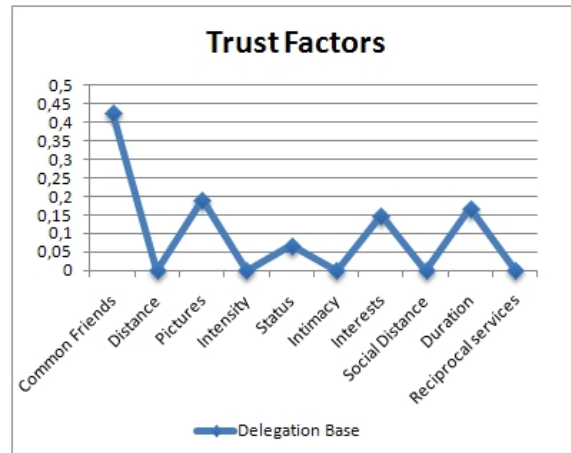


Figura 3.15: Factores Trust

- Música.
 - Series.
 - Películas.
- Comentarios en el muro.

Para realizar el cálculo del *trust* nos hemos basado en los estudios de (Quijano-Sánchez et al., 2010), (Quijano-Sánchez et al., 2011a) y (Quijano-Sánchez et al., 2011b) como se refleja en la Figura 3.15. Las conclusiones obtenidas tras analizar estos estudios son que a la hora de calcular la afinidad entre dos usuarios no todos los elementos tienen la misma importancia e incluso, tal y como demuestran los experimentos se realizaron, algunos tienen un peso tan bajo a la hora de realizar el cálculo que puede incluso omitirse. Para llevar a la práctica el cálculo del *trust* en *Facebook*, es importante tener en cuenta que algunos de estos elementos nos son fáciles de cuantificar y que la obtención de los mismos estará limitada por las capacidades que nos otorguen las APIs de *Facebook*.

En la actualidad conviven dos APIs en esta Red Social. Una de ellas es la llamada *OldAPI*, que como su nombre indica, está cayendo en desuso. Pero esta API es la que nos va a permitir manejar toda la información de la cuenta de un usuario, facilitándonos a través de las funciones incluidas en la misma el acceso a los elementos que queremos extraer. La forma de obtener esta información se basa en consultas a la base de datos del propio *Facebook*. Estas consultas utilizan un lenguaje propio *Facebook* denominado FQL, que es una simplificación del conocido lenguaje SQL; y que nos permite realizar consultas sencillas para la obtención de datos.

Ejemplo de sintaxis de una consulta FQL:

```
SELECT uid, name, picSquare FROM user WHERE uid = me() OR uid
IN (SELECT uid2 FROM friend WHERE uid1 = me())
```

El principal problema está en que algunos de los elementos propuestos (como la distancia de una amistad) no se pueden obtener utilizando las APIs de *Facebook*, por lo que debemos suprimirlos a la hora de realizar el cálculo del *trust*. Finalmente los elementos elegidos para el cálculo del *trust* han sido:

- Número de mensajes en el muro (de entre los 100 últimos del usuario).
- Número de amigos en común.
- Número de fotos en común.
- Intereses comunes:
 - Series.
 - Películas.
 - Música.

Después de realizar una serie de experimentos con usuarios reales de los que conocíamos la confianza real que tenían entre ellos, ajustamos los pesos para que reflejaran con exactitud la confianza que existía entre ellos, obteniendo los siguientes valores:

$$\text{Trust} = \text{amigoscomunes} * 0,3 + \text{fotoscompartidas} * 0,3 + \text{comentariosMuro} * 0,25 + \text{InteresesComunes} * 0,15 \quad (3.2)$$

Observamos que los resultados obtenidos de los pesos son coherentes ya que los factores más importantes en una amistad son el número de amigos comunes y las fotos en las que aparecen juntos los usuarios. El número de comentarios del muro también es un factor importante a la hora de reflejar el grado de amistad, aunque con un peso menor que el de los dos factores anteriores. En un segundo plano queda el número de intereses comunes, que si bien es importante, no es un factor fundamental a la hora de medir la intimidad de la relación entre dos personas.

El cálculo del *trust* se realiza cada vez que un usuario decide aceptar la invitación para unirse a un evento. Es entonces cuando se llama a la función que se encarga de recorrer la tabla que almacena la lista de asistentes al evento en concreto, y de calcular el *trust* del usuario que aceptó unirse al evento

con cada uno de los usuarios que previamente indicaron que asistirían, para seguidamente almacenarlo en la tabla *trust*.

Fundamento teórico del módulo de confianza

Las investigaciones actuales indican que las personas tienden a confiar más en las recomendaciones que provienen de individuos en los que confían (amigos) que en aquellas basadas en *ratings* anónimas (Sinha y Swearingen, 2001). Este elemento social es incluso más importante cuando se realizan recomendaciones a grupos de personas donde los usuarios tienen que ponerse de acuerdo en un artículo para todo el grupo. Esta clase de recomendaciones sigue un proceso de argumentación, donde cada usuario defiende sus preferencias y rebate las opiniones de otros. Aquí es cuando los usuarios deben ceder y modificar sus preferencias para llegar a una solución común, la confianza entre usuarios se convierte en un factor crucial.

Es necesario resaltar que la confianza está relacionada con la intensidad del vínculo. Varios trabajos han concluido que estos dos elementos son conceptualmente diferentes pero que existe una correlación entre ellos (Levin et al., 2004). Por tanto el módulo de confianza también debe verse como una forma de medir la intensidad de los vínculos existentes entre los individuos.

Es sabido que los usuarios de las redes sociales publican en sus perfiles una gran cantidad de información personal. Este conocimiento se puede analizar para calcular la confianza con otros usuarios: distancia en la red social, número de comentarios entre amigos, gustos e intereses, información personal, fotos, juegos, duración de la amistad... (Golbeck, 2006b), (Gilbert y Karahalios, 2009). Tenemos que ser conscientes que la estimación de un valor de confianza entre dos personas puede variar entre diferentes redes sociales, y que independientemente de la Red que sea, esta estimación dista mucho de una valoración directa.

3.2.3.3. Almacenamiento de la Satisfacción

En este módulo se guardan todas las recomendaciones que se han hecho para cada usuario y cada grupo. Como disponemos de un sistema de recomendaciones con memoria nos permite evitar repetir recomendaciones previas y a la vez asegurar un cierto grado de justicia. Si un componente del grupo acepta una propuesta que no le agrada, la próxima vez sus opiniones tendrán más peso y así, a la larga, tendremos todas las opiniones de los miembros del grupo igualmente satisfechas.

El almacenamiento de la satisfacción consiste en una base datos que almacena un valor que representa la satisfacción de cada usuario con cada grupo al que pertenece. Este valor se puede aplicar después en las fórmulas de recomendación para modificar el resultado y otorgar una mayor influencia a aquellos usuarios que tienen el nivel de satisfacción más bajo. El valor de la satisfacción de cada usuario se modifica cada vez que el usuario obtiene una recomendación para un grupo al que pertenece. Este módulo permite la implementación de *recomendadores adaptativos*.

Una vez que el usuario accede al evento tendrá la posibilidad votar las tres películas que el recomendador grupal ha seleccionado para el grupo. Cuando el usuario realiza esta votación se almacenan los valores de las votaciones en una tabla de la base de datos que refleja cuan de satisfecho está el usuario con las propuestas obtenidas. Más tarde si se desea hacer una valoración de la satisfacción global o individual de cada uno de nuestros usuarios bastaría con analizar las votaciones dadas en esta tabla y compararlo con la película final propuesta para el grupo.

Fundamento teórico para el almacenamiento de la Satisfacción

Después de aplicar los factores de personalidad y confianza debemos asegurar un cierto grado de satisfacción entre todos los miembros del grupo. Hasta ahora nos hemos centrado en situaciones donde el recomendador realiza la recomendación una única vez. Pero es normal pensar que un grupo esperará poder usar el sistema varias veces y obtener de esta manera una mayor muestra de recomendaciones.

Sin embargo, utilizando sólo estas nuevas técnicas de recomendación acabaríamos siempre favoreciendo a los mismos usuarios (ya sea porque tiene personalidades más fuertes o porque existe mayor proximidad entre ellos), y podríamos acabar teniendo una situación donde hubiese varios usuarios insatisfechos, ya que tenemos sus opiniones menos en cuenta por el bien del grupo. Para ser capaces de evitar situaciones donde los niveles de insatisfacción del grupo sean muy altos, debemos tener en cuenta un cierto grado de justicia. Por ejemplo, se puede dar el caso en el que una recomendación es muy prometedora para el grupo en general, pero sin embargo uno de los usuarios puede verse muy disgustado con esa recomendación. Sería deseable que en las futuras recomendaciones se le favoreciese a este componente del grupo de alguna manera para que éste pudiese alcanzar un nivel adecuado de satisfacción.

Para resolver este asunto, proponemos utilizar una memoria de las re-

comendaciones ya realizadas. El tener recomendaciones con memoria significa que somos capaces de crear un sistema que recuerda todas las recomendaciones que se le han realizado a lo largo del tiempo a un grupo. Creemos que este es un paso necesario cuando se quiere proporcionar una colección de recomendaciones justas. De esta forma, si un miembro del grupo acepta una propuesta en la que no está interesada, la próxima vez tendrá algún tipo de prioridad en el proceso de recomendación.

3.2.3.4. Test de preferencias

El objetivo del test de preferencias es conocer los gustos del usuario en cuanto a películas se refiere. A la hora de hacer la recomendación individual se tendrán en cuenta estas las votaciones atendiendo al género, a los actores, el año de la película...

Por ejemplo, si has votado *el Ultimatum de Bourne* con 4.5 puntos como se muestra en la Figura 3.16, se tendrá en cuenta que has dado una votación muy alta a esa película. Y fijándonos en su género, en los actores... como se comentó previamente, se realizará la recomendación individual más óptima posible.

Observando otra vez la Figura 3.16, podemos ver una la barra superior que muestra el número de películas que lleva votadas el usuario y el número de películas totales que se pueden votar. Hay que tener un mínimo de 20 películas votadas para poder realizar cualquier acción sobre la aplicación *HappyMovie*, ya que con esta restricción tiene suficiente información para analizar satisfactoriamente. En otro caso la recomendación no sería lo suficientemente ajustada a los gustos del usuario. Superado este número de películas votadas, la barra cambia de color del rojo al verde, como se muestra en la Figura 3.17.

Este test permite no votar una película ya sea porque no te apetece votarla o porque no la has visto. La siguiente vez que se muestre dicha película saldrá sin ninguna puntuación. Del mismo modo, si ya fue votada anteriormente saldrá con la puntuación otorgada en su momento, dando la posibilidad de volver a votarla, sustituyendo su puntuación anterior por la nueva.

3.2.3.5. Recomendador individual

La función de este recomendador es analizar las preferencias de cada usuario respecto a las películas que actualmente se encuentran en la cartel-

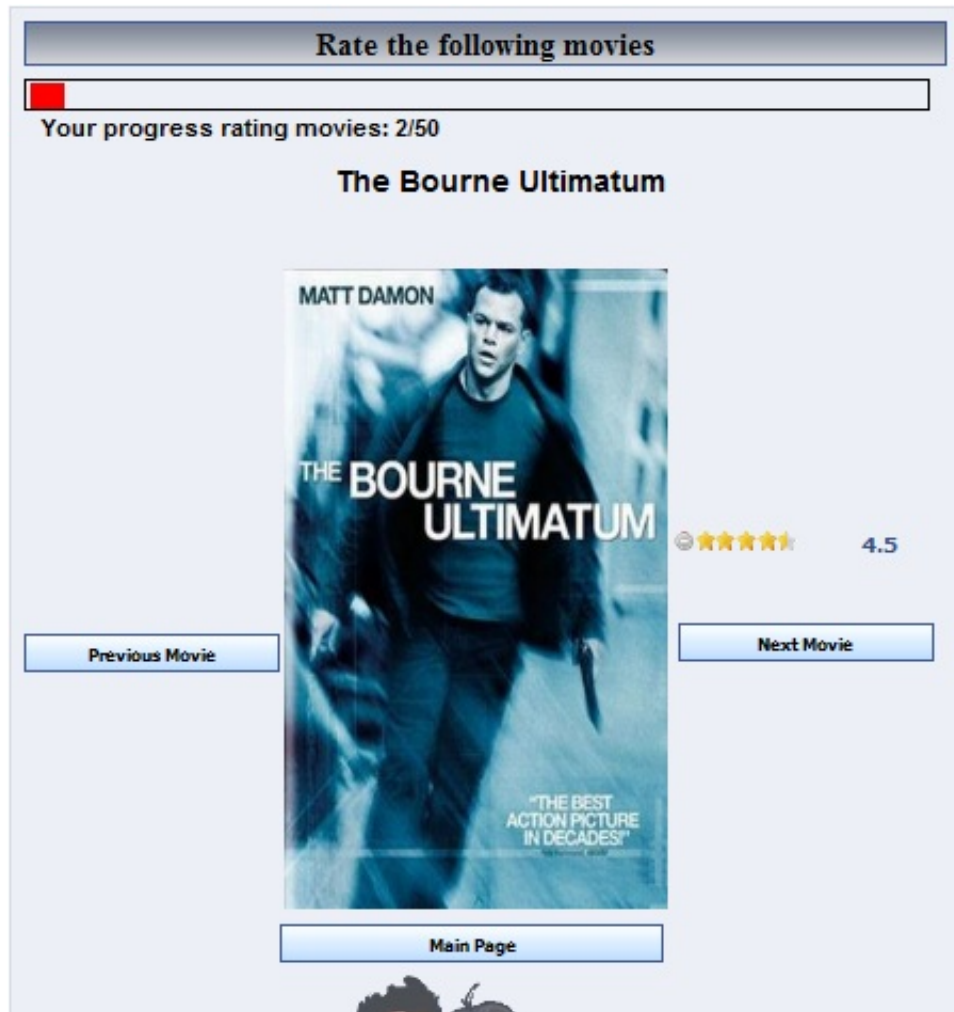


Figura 3.16: *Interfaz test de preferencias*

Figura 3.17: *Interfaz test de preferencias*

era y que se actualizan cada semana. Las preferencias están almacenadas en la base de datos, son datos que se han obtenido del test de preferencias votando al menos 20 películas para poder obtener una recomendación fiable.

Para hacer las recomendaciones se tienen en cuenta los siguientes factores:

- Género.
- Año.
- Nacionalidad de la película.
- Duración.
- Actores.
- Directores.
- Calificación por edad.

Además de hacer la recomendación en función a la similitud de todos estos campos, se les ha asignado diferentes pesos al considerar que hay ciertos factores que son más importantes que otros a la hora de recomendar. Los pesos de los campos, siendo la suma total igual a uno, son los siguientes:

- PESO año = 0.02
- PESO duracion = 0.02
- PESO calificacion = 0.1
- PESO nacionalidad = 0.1
- PESO actores = 0.26
- PESO directores = 0.2
- PESO genero = 0.3

Nuestro recomendador individual es un recomendador basado en casos. Tomamos como casos las preferencias de cada usuario, que inicialmente se encuentran almacenadas en un archivo, y que se actualizarán automáticamente cuando se voten más películas en el test de preferencias.

Teniendo almacenadas las preferencias de cada usuario de *HappyMovie* se realizan la recomendación individual cuando un usuario acepta formar parte de un evento. El resultado final de una recomendación individual se muestra en la Figura 3.18.

En la recomendación almacenamos:

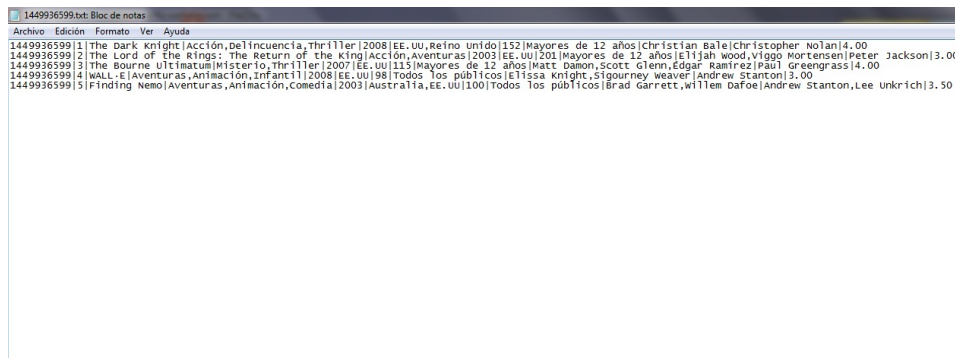


Figura 3.18: Archivo test de preferencias

- Id del usuario.
- Id de la película en la tabla Movies info de la BBDD.
- Título de la película.
- Género.
- Año.
- Duración (en minutos).
- Calificación por edad.
- Actores.
- Directores.
- Rating otorgado a la película por el usuario

Las películas actualmente en cartelera se guardan en un archivo en el que se crea una lista de queries, en el que cada query contiene los mismos campos anteriormente mencionados a excepción del rating otorgado a la película. El resultado se muestra en la Figura 3.19.

Una vez tenemos todos los datos en su formato correcto realizamos la recomendación individual, cuyo resultado se devuelve en un archivo de texto con el siguiente formato:

- Id del usuario
- Id de la película
- Rating

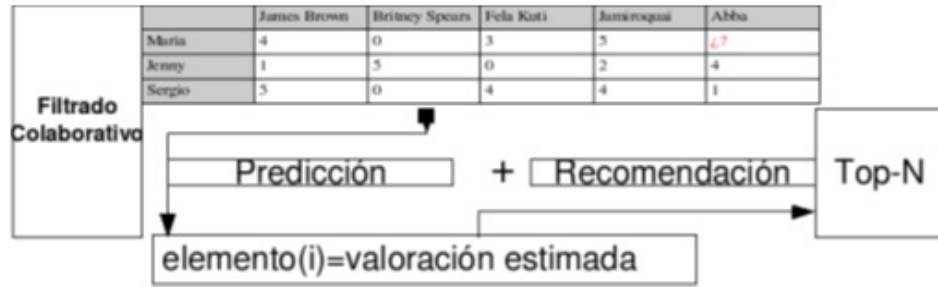


Figura 3.19: Archivo .xml con las películas de la cartelera

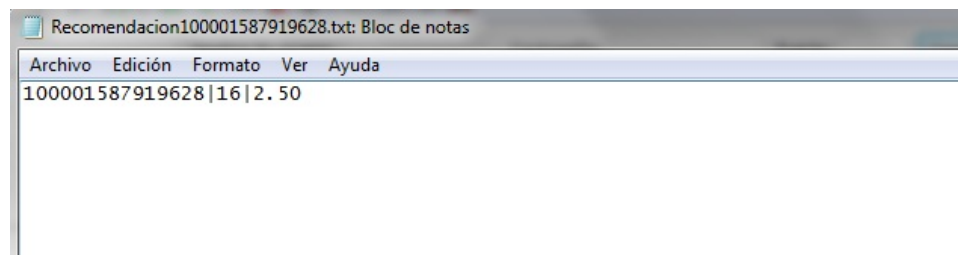


Figura 3.20: Archivo resultado del recomendador individual

Así se muestra en la Figura 3.20.

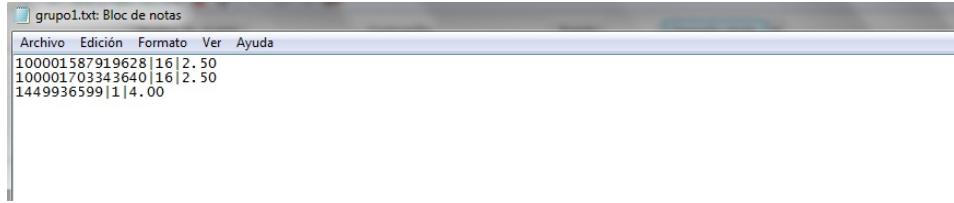
El recomendador individual se ejecuta cada vez que se añaden nuevas votaciones en el test de preferencias y cuando la cartelera de los cines se actualiza.

3.2.3.6. Recomendador de grupo

El recomendador grupal reúne todas las recomendaciones individuales de los asistentes al evento, y da una recomendación final en base a dichas recomendaciones y al trust que une a todos los usuarios entre sí.

Cuando llega la fecha del *deadline* se calcula si los asistentes al evento son muy parecidos a otros grupos ya creados, teniendo como condición que si el 60 % de los participantes no pertenecen a otro grupo se crea uno nuevo con todos los asistentes a dicho evento. Y en otro caso, se considera que la similitud con otro grupo es alta y la recomendación se considera de ese grupo.

Lo primero que hace el recomendador grupal es reunir en un único archivo todas las recomendaciones individuales como se observa en la Figura 3.21. El recomendador se ejecuta diariamente teniendo como condición necesaria que o bien sea la primera vez, o bien el número de asistentes al evento haya



Archivo	Edición	Formato	Ver	Ayuda
100001587919628	16	2.50		
100001703343640	16	2.50		
1449936599	1	4.00		

Figura 3.21: Recomendaciones individuales de todos los asistentes al evento

variado, igual que el individual, actualizándose dichas recomendaciones hasta el día del *deadline* que realiza la última recomendación.

Además, se necesita:

- Tabla de personalidad. Contiene todos los usuarios asistentes con su personalidad, datos obtenidos al realizar el test de personalidad al comienzo de la aplicación *HappyMovie*.
- Tabla Trust. Esta tabla está formada por todas las valoraciones **trust** de todos los asistentes al evento.

Toda esta información juega un papel relevante en la decisión final de las tres películas que se recomiendan.

Funciones de agregación:

Se ha ideado un amplio conjunto de funciones de agregación para la combinación de las preferencias individuales (Masthoff, 2004), siendo las estrategias de *Maximizar la satisfacción media* y *Minimizar la miseria* las más comúnmente usadas. En esta sección explicamos algunas de las funciones de agregación aplicables al método de recomendación grupal.

Algunas de estas estrategias se explican a continuación:

- **Maximizar la satisfacción media:** Utiliza un recurso aritmético frecuente, la media, es un método para derivar la tendencia central de una muestra en el espacio. Calcula la media de los ratings estimados para cada miembro del grupo. La función que implementa esta estrategia es:

$$gpred(G, i) = \frac{1}{|G|} \sum_{u \in G} pred'(u, i) \quad (3.3)$$

Donde $pred'(u, i)$ es el rating estimado para cada usuario u , y cada artículo i . $gpred(G, i)$ es el rating final estimado para el grupo para el

producto i .

- **Recuento Borda:** Es un proceso electoral propuesto por el matemático francés Jean-Charles Borda en 1770. Según esta metodología, los candidatos son ordenados según las preferencias de cada elector; en el recuento, a cada posición de la ordenación le es atribuida una puntuación: 1 punto para el último clasificado, 2 puntos para el penúltimo, 3 para el antepenúltimo etc. Este sistema lleva en consideración no sólo la primera opción de cada elector, sino también las otras, de modo que no siempre el candidato colocado más veces en primer lugar es el vencedor.

$$\begin{aligned}
 gpred(G, i) &= \sum_{u \in G} bs(u, i) \\
 bs(u, i) &= pos(pred'(u, i), OL) \\
 OL &= \{pred'(u, i_1), pred'(u, i_2), \dots, pred'(u, i_n)\} \\
 &\quad \text{where } pred'(u, i_p) \leq pred'(u, i_{p+1})
 \end{aligned} \tag{3.4}$$

Donde $bs(u, i)$ es la puntuación asignado para cada elemento que el usuario u haya votado.

- **Regla Copeland:** En esta estrategia se ordenan las diferentes alternativas según el número de victorias que un elemento tiene sobre los otros (Copeland, 1951). Es un buen procedimiento que evita los problemas que existen en los ciclos de votaciones (Klamler, 2005).

$$\begin{aligned}
 gpred(G, i) &= \sum_{j \in Catalog, j \neq i} cs(i, j) \\
 cs(i, j) &= \begin{cases} +1 & \text{if } gana(i, j) > pierde(i, j) \\ -1 & \text{if } gana(i, j) < pierde(i, j) \\ 0 & \text{a.o.c.} \end{cases} \\
 gana(i, j) &= |pred'(u, i) > pred'(u, j)|, \forall u \in G \\
 pierde(i, j) &= |pred'(u, i) < pred'(u, j)|, \forall u \in G
 \end{aligned} \tag{3.5}$$

- **Voto aprobatorio:** Es un sistema de votación utilizado para las elecciones en el cual un votante puede votar por todos los candidatos que desee. Se suele utilizar en los casos de elecciones con un único ganador, si bien también puede extenderse a elecciones con múltiples ganadores, aunque con implicaciones matemáticas distintas. El voto aprobatorio es una forma primitiva de voto valorativo, en el cual se permite una

valoración algo más compleja de los candidatos (por ejemplo, puntuándoles).

Sin embargo, en el voto aprobatorio, sólo cabe votar “aceptación” o “no aceptación”. Con esta estrategia cada usuario puede votar (o aprobar) cuantos elementos como desee. El artículo ganador será aquel que obtenga más votos.

$$\begin{aligned} gpred(G, i) &= \sum_{u \in G} as(u, i) \\ as(u, i) &= \begin{cases} 1 & \text{if } pred'(u, i) \geq \delta \\ 0 & \text{a.o.c.} \end{cases} \end{aligned} \quad (3.6)$$

- **Minimizar la miseria:** Esta estrategia contempla la idea de que, incluso si la media global de los ratings sobre un artículo es alta, una solución que deja a un miembro del grupo muy insatisfecho se debe descartar. En ella se considera que un grupo es tan feliz como lo es su miembro más infeliz, con una solución dada. La lista final de los ratings es el mínimo de cada rating individual para un artículo. Una desventaja de este método es que aunque a la mayoría del grupo les gusta mucho un artículo con que a una persona no le guste, ese producto nunca será elegido.

$$gpred(G, i) = \min_{u \in G} pred'(u, i) \quad (3.7)$$

- **Estrategia Más deseado:** Es la estrategia contraria a la anterior, Minimizar la miseria, selecciona el rating más alto dado a cada elemento, y de ahí se forma la lista final de ratings estimados.

$$gpred(G, i) = \max_{u \in G} pred'(u, i) \quad (3.8)$$

- **Media sin Miseria:** asigna como estimación la media de todos los ratings dadas por cada individuo. En esta estrategia existe una diferencia y es que no se tendrán en cuenta aquellos artículos que tengan un rating de alguno de los usuarios inferior a una cierta cota.

$$gpred(G, i) = \frac{1}{|G|} \sum_{u \in G | pred'(u, i) > \delta} pred'(u, i) \quad (3.9)$$

Una vez descritas todas las funciones de agregación que se pueden utilizar para combinar las estimaciones individuales describimos a continuación cómo modificar los ratings individuales estimados por medio de factores sociales. En ella presentamos dos métodos que hemos desarrollado en nuestra arquitectura ARISE, con ellos somos capaces de mejorar las recomendaciones grupales existentes por medio de factores de personalidad y confianza.

Modificación de las predicciones individuales con factores sociales:

Nuestros métodos de recomendación consisten en evaluar los diferentes comportamientos que la gente tiene en un proceso de toma de decisión. Para conseguir esto modificamos las predicciones obtenidas con el recomendador individual con la personalidad de los usuarios y los factores de confianza. De esta manera no todas las predicciones se tienen en igual consideración. Hemos creado diferentes métodos para calcular el nuevo rating individual, $pred'(i, u)$, que es el que se aplica a la Ecuación final de recomendación.

- **Método basado en delegación:** El enfoque basado en delegación sigue las ideas presentadas en Golbeck (2006a), donde el rating individual de una persona está basado en los ratings dados por otros usuarios. La idea detrás de este enfoque es que la opinión del usuario se crea basándose en las opiniones de sus amigos. Este promedio de opiniones se calibra dependiendo del nivel de confianza con cada amigo.

Adicionalmente, la personalidad de cada amigo también se tiene en cuenta, modificando así la opinión base. El rating basado en delegación (*delegation-based rating*) intenta simular el siguiente comportamiento: cuando se decide qué elemento elegir entre un grupo de usuarios, preguntaremos a la gente con la que tengamos más confianza. Por tanto, también tendremos en cuenta su personalidad, y le daremos más importancia a unas preferencias que a otras.

El rating basado en delegación se calcula del siguiente modo:

$$dbr(u_i, m) = \frac{\sum_{u_j \in G \wedge j \neq i} (t_{ij} \cdot (ir_{j,m} + CMW_j))}{\sum_{u_j \in G \wedge j \neq i} t_{ij}} \quad (3.10)$$

donde t_{ij} es la confianza existente entre el usuario i y el j .

- **Método basado en influencia:** El enfoque de los ratings basados en influencia simula la influencia que tiene cada amigo en una deter-

minada persona. En vez de crear una nueva preferencia, parte de la suposición de que el usuario modificará sus preferencias por un elemento dado, dependiendo de las preferencias de sus amigos por ese mismo. Por ejemplo, si nuestro rating para un elemento es de 3 y nuestro amigo tiene un rating para el mismo elemento de 5, podríamos modificar nuestro rating a 4. Dependiendo de la confianza con este amigo en concreto decidiremos el nivel de variación de nuestro rating (i.e. daremos 3.5 si es baja, o 4.5 si la confianza es alta).

Además la variación del rating también dependerá de nuestra propia personalidad. Si tenemos una personalidad muy fuerte no estaremos dispuestos a cambiar nuestra valoración, pero si en cambio tenemos una personalidad débil, fácilmente nos podrán influenciar las opiniones de otros. Combinando estos dos factores obtenemos la siguiente formula:

$$ibr(u_i, m) = ir_{i,m} + (1 - CMW_i) \cdot \frac{\sum_{u_j \in G \wedge j \neq i} (t_{ij} \cdot (ir_{j,m} - ir_{i,m}))}{|G|} \quad (3.11)$$

donde t_{ij} es la confianza existente entre el usuario i y el j y $|G|$ es el número de componentes del grupo.

Los dos métodos aquí expuestos, representan dos posibles maneras de realizar las combinaciones de las predicciones individuales (ya modificadas con los valores de personalidad y confianza) para proporcionar una recomendación al grupo.

3.2.3.7. Bases de datos

Como en la mayoría de las aplicaciones actuales, es fundamental la utilización de bases de datos para poder almacenar y acceder de forma eficiente a toda la información que debe usar nuestra aplicación. Para la base de datos de nuestra aplicación el diseño lo hemos dividido en cuatro partes fundamentales.

La primera parte es necesaria para el funcionamiento del módulo *Web Crawler* (Sección 3.2.3.8). Utiliza cinco tablas:

- Actúan.
- Película.
- Dirigen.

- Provincia.
- PeliProvincia.

cuyas tablas están reflejadas en la Figura 3.23. Están gestionadas con un motor *INNODB*. En ellas almacenaremos toda la información obtenida semanalmente de las carteleras.

La segunda parte se refiere al módulo test de personalidad. Utiliza la siguiente tabla:

- Usuarios.

ver Figura 3.22. Está gestionada con un motor *MYISAM*. En ella se almacena el resultado del cálculo de la personalidad de cada usuario.

La tercera parte se usa para el funcionamiento del módulo test de preferencias. Utiliza las tres tablas siguientes:

- rated movies
- movies info
- users

Estas tablas están reflejadas en la Figura 3.22, que utilizan un motor *MYISAM*.

La cuarta y última parte son las tablas necesarias para el funcionamiento de los módulos recomendadores. Estos módulos, debido a su complejidad, necesitan acceder prácticamente a todas las tablas de nuestra aplicación. Es decir, a todas las tablas del módulo *Web Crawler* como a todas las tablas de la Figura 3.23 (a excepción de las tablas evento, invitación, votes y scores).

El resto de las tablas que no han sido mencionadas se utilizan para el funcionamiento básico de la aplicación: almacenamiento de usuarios, eventos, invitaciones...

3.2.3.8. Web Crawler

Web Crawler es la herramienta mediante la cual rescatamos la información de la cartelera de todas las provincias de España. Toda la información que rescatamos proviene de la Guía del Ocio³, que contiene información detallada de toda la información que necesitamos:

³<http://www.guiadelocio.com>

- Título de la película
- Directores
- Actores
- Año
- Género
- Duración
- Calificación
- Nacionalidad
- Cartel de la película

Toda esta información es la correspondiente a una película, rescatando esta información de todas las películas que están actualmente en la cartelera de todos los cines de España, organizados por provincias (tabla *PeliProvincia* de la Figura 3.23), de tal modo que sabemos que película es proyectada en qué provincia.

Esta herramienta se actualiza cada vez que la cartelera de los cines cambia, es decir, todos los viernes. Esta importante funcionalidad nos permite tener actualizada la Base de Datos en todo momento sin necesidad de estar pendientes de la cartelera actual.

Para el correcto funcionamiento de *Web Crawler* se necesitan cinco tablas (Figura 3.23) donde la información queda debidamente guardada y perfectamente organizada, de tal modo que cuando el usuario crea un evento en la ciudad que él ha elegido, podemos rescatar toda la información correspondiente a las películas proyectadas en esa ciudad, teniendo fácil acceso a los datos de las películas (anteriormente enumerados), ya que se necesitarán para poder recomendar al usuario la película que más casa con sus gustos (Test de preferencias, Sección 3.2.3.4), dicha recomendación se recalcula cuando *Web Crawler* se ejecuta la herramienta.

Organización Web Crawler:

- *Película*. Esta tabla contiene la información sobre cada película que se proyecta en España. Esta tabla contiene los siguientes campos:
 - ID de la película
 - Título de la película
 - Año

- Género
 - Duración
 - Calificación
 - Nacionalidad
 - Url
- *Dirigen*. Esta tabla contiene los nombres de todos los directores de cada película.
 - IDPelícula
 - Director

A través del ID de la película relacionamos el nombre de los directores con la información de la película.

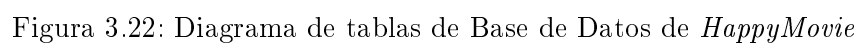
- *Actuan*. De forma análoga a *Dirigen*, aquí se almacenan los nombres de los actores principales que aparecen en la película.
 - IDPelícula
 - Actor
- *PeliProvincia*. Esta tabla permite saber qué película se proyecta en qué provincia, ya que una misma película se puede proyectar en diferentes provincias.
- *Provincia*. Esta tabla asigna un ID a cada provincia de España, de tal modo que al referirnos en la tabla *PeliProvincia* usaremos dicho ID en vez del nombre.

3.3. Descripción funcional del sistema. Casos de uso

A continuación, se exponen los casos de uso que hemos desarrollado a lo largo de nuestro proyecto:

3.4. Funcionamiento global de *HappyMovie*

Cuando el usuario accede a la página principal de la aplicación, en primer lugar se comprueba que haya iniciado sesión en *Facebook* y, en caso contrario, se redirige a la página de inicio de sesión por defecto de *Facebook*. La primera vez que entras a la aplicación *HappyMovie*, de la misma manera que en el resto de aplicaciones de *Facebook* requerirá que aceptes los permisos para poder tener acceso a los datos personales del usuario.



Objetivo	Abrir la aplicación en <i>Facebook</i> .
Prerrequisitos	Tener una cuenta <i>Facebook</i> . Tener abierta tu cuenta de <i>Facebook</i> en ese momento.
Actores	Usuario
Acciones	1. Hacer clic con el ratón en la aplicación <i>HappyMovie</i> .
Comentarios	La aplicación <i>HappyMovie</i> se encontrará situada entre las otras aplicaciones que ofrece <i>Facebook</i> . Una vez accede a la aplicación, al usuario se le muestran dos cosas: 1. Los eventos ya creados a los que ha indicado que asistirá. 2. Un botón “Crear nuevo evento” que permite crear un nuevo evento para ir al cine.

Tabla 3.2: Caso de Uso 1. Acceder a la aplicación.

Objetivo	Mostrar la pantalla principal de la aplicación al usuario.
Prerrequisitos	Que el usuario le haya dado al botón de acceder a la aplicación. Que el usuario acepte las condiciones de uso de la aplicación (permitir leer datos).
Actores	La aplicación.
Acciones	Comprobar si es la primera vez que el usuario accede a la aplicación: 1. La primera vez: darle de alta en la base de datos, y mostrar el botón de “Crear nuevo evento”. Mostrar otro botón que indique “Hacer test de personalidad”. Mostrar un último botón que indique “Hacer test de preferencias”. 2. Si NO es la primera vez, leer de la base de datos los eventos existentes en ese momento a los que ha indicado que asistirá y mostrárselos en una lista. Además mostrar el botón de “Crear nuevo evento”.
Comentarios	La base de datos de usuarios contiene el identificador del usuario (IdU) que es el número que asigna <i>Facebook</i> a cada usuario. dicho número es único y será nuestra clave principal de la tabla Users. Dicha tabla contendrá también la personalidad de cada usuario (Personality) y su nombre en <i>Facebook</i> (Nombre).

Tabla 3.3: Caso de Uso 2. Abrir la aplicación .

Objetivo	Crear un nuevo evento ir al cine.
Prerrequisitos	Haber contestado a los test de personalidad y preferencias. En caso de no haber realizado estos tests, no realizar ninguna acción al pulsar el botón e indicar debajo del botón mediante un cartel "Necesita rellenar X para realizar esta acción". Donde X será "los dos tests de personalidad y preferencias", "el test de personalidad" o "el test de preferencias" dependiendo de los que le falten por completar al usuario.
Actores	Usuario.
Acciones	<ol style="list-style-type: none"> 1. Hacer clic con el ratón en el botón de "Crear nuevo evento". 2. Elegir un nombre para el evento. 3. Indicar la fecha en la que los usuarios ya no podrán apuntarse al evento. 4. Indicar la fecha en la que tendrá lugar la realización del evento. 5. Indicar la ciudad donde tendrá lugar el evento (elegir mediante un desplegable). 6. Dar en el botón de "Finalizar".
Comentarios	<ol style="list-style-type: none"> 1. Una vez el usuario finaliza la creación del evento, se le muestra la pantalla principal del evento ->Cu7. 2. Se modifica la tabla "Eventos", que guarda la información relativa al evento, cuando y donde tendrá lugar y el deadline para invitar amigos. Evento(IdEvento, Nombre,ciudad, fecha, deadlineInvitación, creador). 3. Se actualiza la tabla "Attendance", que guarda los usuarios asistiendo a cada evento. Attendance(IdU,IdEvento), con el usuario creador del evento.

Tabla 3.4: Caso de Uso 3. Crear un nuevo evento.

Objetivo	Invitar amigos a unirse a un evento en el que el usuario participa.
Prerrequisitos	El usuario debe haber indicado que asistirá, ya sea porque lo ha creado él, o porque ha aceptado una invitación.
Actores	Usuario.
Acciones	1. Hacer clic con el ratón en el botón “Invitar amigos”. 2. Seleccionar de entre toda la lista de amigos de <i>facebook</i> del usuario a los que desea invitar.
Comentarios	

Tabla 3.5: Caso de Uso 4. Invitar amigos al evento.

Objetivo	La aplicación notifica a un usuario que un amigo suyo le invita a unirse a un evento a través de <i>HappyMovie</i> .
Prerrequisitos	Que el usuario que recibe la notificación haya sido invitado a un evento ya creado.
Actores	La aplicación.
Acciones	La aplicación envía al usuario una notificación de que un amigo suyo le invita a unirse a un evento a través de <i>HappyMovie</i> .
Comentarios	Esta notificación aparecerá en el apartado “Solicitudes” del perfil de <i>facebook</i> , junto con las otras solicitudes que el usuario tenga de otras aplicaciones, eventos, amistades..etc. En la solicitud aparecerá el nombre de la aplicación junto con el nombre del amigo que ha enviado la solicitud. Además aparecerán dos botones uno de “Aceptar” y otro de “Ignorar”.

Tabla 3.6: Caso de Uso 5. Mostrar solicitud de unirse a un evento.

Objetivo	El usuario indica que acepta unirse al evento.
Prerrequisitos	Haber contestado a los test de personalidad y preferencias. En caso de no haber realizado estos tests, no realizar ninguna acción al pulsar el botón e indicar debajo del botón mediante un cartel “Necesita rellenar X para realizar esta acción”. Donde X será “los dos tests de personalidad y preferencias”, “el test de personalidad” o “el test de preferencias” dependiendo de los que le falten por completar al usuario.
Actores	Usuario.
Acciones	<ol style="list-style-type: none"> 1. Hacer clic con el ratón en el botón “Aceptar” en la solicitud de unirse al evento de <i>HappyMovie</i>. 2. Al usuario se le mostrará la pantalla principal de <i>HappyMovie</i>. 3. Calcular el trust con todos los usuarios que asistirán al evento en ese momento. Y almacenarlo en la base de datos.
Comentarios	<ol style="list-style-type: none"> 1. La aplicación lo registrará, si es la primera vez que utiliza la aplicación antes de poder acceder al evento deberá completar los dos test de personalidad y preferencias, la aplicación lo incluirá en la base de datos. Una vez ya haya realizado estos pasos o si no es la primera vez que utiliza <i>HappyMovie</i> se le incluirá en la lista de usuarios atendiendo al evento y se actualizarán los resultados de la recomendación. 2. La base de datos se actualizará después de haber calculado la confianza del nuevo usuario con cada uno de los asistentes hasta ese momento. La tabla será la siguiente: Trust(IdU1, IdU2, trust), donde IdU1 es el identificador del usuario1, IdU2 es el identificador del usuario2 y trust es el compute final de la confianza entre ambos. 3. Se actualiza la tabla “Attendance”, que guarda los usuarios asistiendo a cada evento. Attendance(IdU, IdEvento).

Tabla 3.7: Caso de Uso 6. Aceptar unirse a un evento.

Objetivo	Acceder a la pantalla principal de un evento particular en el que el usuario ha decidido participar.
Prerrequisitos	<p>1. Haber contestado a los test de personalidad y preferencias. En caso de no haber realizado estos tests, no realizar ninguna acción al pulsar el botón e indicar debajo del botón mediante un cartel "Necesita rellenar X para realizar esta acción". Donde X será "los dos tests de personalidad y preferencias", "el test de personalidad" o "el test de preferencias" dependiendo de los que le falten por completar al usuario.</p> <p>2. Haber dado a aceptar la invitación del evento o haber creado el evento.</p>
Actores	Usuario.
Acciones	<p>1. Hacer clic con el ratón en la aplicación <i>HappyMovie</i></p> <p>2. Hacer clic con el ratón en el evento al que se desea acceder.</p> <p>3. Se mostrará la página principal de ese evento en concreto.</p>
Comentarios	<p>1. Cu7 A) No ha finalizado el deadline de unirse al evento: La pantalla principal del evento incluye, una lista de los usuarios que han aceptado unirse al evento hasta ese momento, la posibilidad de borrarse del evento, la posibilidad de invitar nuevos amigos al evento, la fecha y localización del evento y la lista de las tres películas recomendadas para los usuarios que se han unido al evento hasta ese momento.</p> <p>2. Cu7 B) Ha finalizado el deadline de unirse al evento: La pantalla principal del evento incluye, una lista de los usuarios que han aceptado unirse al evento, la fecha y localización del evento y la lista de las tres películas recomendadas para los usuarios definitiva y la posibilidad de votar cada una de esas tres películas.</p> <p>3. En la pantalla principal de la aplicación se ve un listado de todos los eventos a los que el usuario ha aceptado participar.</p>

Tabla 3.8: Caso de Uso 7. Acceder a un evento en el que el usuario ya está incluido .

Objetivo	Borrar la participación del usuario en un determinado evento.
Prerrequisitos	1. Estar como participante en ese evento. 2. Estar en la pantalla principal del evento.
Actores	Usuario.
Acciones	1. Hacer clic con el ratón en el botón “No asistiré”. 2. El usuario desaparece de la lista de personas atendiendo al evento. 3. El evento ya no le aparecerá al usuario en la lista de eventos en la pantalla principal de <i>HappyMovie</i> . 4. Se actualiza la recomendación para el grupo.
Comentarios	

Tabla 3.9: Caso de Uso 8. Borrarse de un evento.

Objetivo	Contestar al test de personalidad y guardar cada una de las respuestas en la base de datos.
Prerrequisitos	Acceder a la pantalla principal de <i>HappyMovie</i> . No haber rellenado este test con antelación, en cuyo caso el botón de “Hacer test de personalidad” no aparecerá más en la pantalla principal de la aplicación.
Actores	Usuario.
Acciones	1. Hacer clic con el ratón en el botón “Hacer test de personalidad”. 2. Contestar a todas las preguntas presentadas en el test. 3. Almacenar en la base de datos cada una de las respuestas a las preguntas. 4. Realizar el cómputo final de la personalidad. 5. Guardar en la base de datos el valor final de la personalidad.
Comentarios	1. La base de datos, guarda por cada usuario el valor que identifica su personalidad en la siguiente tabla: Personalidad(IdU, personality) donde IdU es el identificador del usuario y personality es el valor que se ha calculado a través de las respuestas del test de personalidad. 2. La base de datos, guarda también la respuesta a cada de las preguntas del test de personalidad en la siguiente tabla: TestPersonalidad(IdU, respuesta1, respuesta2,..) donde IdU es el identificador del usuario y respuestaN es el valor indicado por el usuario para la pregunta número "N".

Tabla 3.10: Caso de Uso 9. Rellenar test de personalidad.

Objetivo	Contestar al test de preferencias y guardar cada una de las respuestas en la base de datos.
Prerrequisitos	Acceder a la pantalla principal de " <i>HappyMovie</i> ".
Actores	Usuario.
Acciones	<ol style="list-style-type: none"> 1. Hacer clic con el ratón en el botón "Hacer test de preferencias" 2. Contestar a todas las preguntas presentadas en el test. 3. Almacenar en la base de datos cada una de las respuestas a las preguntas.
Comentarios	La base de datos, guarda también la respuesta a cada de las preguntas del test de personalidad en la siguiente tabla: TestPreferencias(IdU, IdMovie, rating) donde IdU es el identificador del usuario, IdMovie es el identificador de la película valorada por el usuario y ratings es la respuesta del usuario para esa película (que tendrá un valor de -1, si no la ha visto o no ha contestado la pregunta)

Tabla 3.11: Caso de Uso 10. Rellenar test de preferencias.

Objetivo	Inhabilitar la posibilidad de invitar más amigos al evento o de aceptar participar en el evento.
Prerrequisitos	Tener el evento creado.
Actores	La aplicación.
Acciones	<ol style="list-style-type: none"> 1. Mostrar la recomendación final para los usuarios que atenderán al evento. 2. Permitir una votación entre las tres películas seleccionadas. 3. Deshabilitar la posibilidad de invitar amigos al evento. 4. Deshabilitar la posibilidad de borrarse del evento 5. Deshabilitar la posibilidad de aceptar una invitación al evento.
Comentarios	

Tabla 3.12: Caso de Uso 11. Finalización del deadline para unirse al evento.

Objetivo	Permitir a los usuarios participantes votar cual de las tres películas propuestas por el recomendador les agrada más.
Prerrequisitos	Que haya finalizado el deadline para apuntarse al evento.
Actores	Usuario.
Acciones	Votar cual de las tres películas propuestas prefiere ver
Comentarios	1. Se contabilizará el número de votos y se guardarán para indicar la película final el día de realización del evento. Además se guardará en la tabla "Películas Vistas" las tres películas y la posición en la que ha quedado cada una en la votación. 2. Cada usuario podrá votar una única vez.

Tabla 3.13: Caso de Uso 12. Votación de películas propuestas .

Objetivo	Eliminar el evento cuando llegue la fecha de finalización.
Prerrequisitos	Tener el evento creado que haya finalizado el deadline para que los usuarios se unan al evento.
Actores	La aplicación.
Acciones	1. Crear un nuevo grupo en el caso de que el 60 % de los participantes no pertenezcan a otro grupo. (Evaluar esto): 2. Si se trata de un nuevo grupo guardar en la tabla "Grupos" el identificador del nuevo grupo. 3. Guardar la película "más votada" en la tabla "PelículasVistas". 4. Guardar las otras dos películas recomendadas e indicar su posición en la votación. 5. Borrar el evento de <i>HappyMovie</i> una vez llegue su fecha de finalización.
Comentarios	1. La tabla "PelículasVistas" guarda las películas que ha visto cada grupo, y las que se le han recomendado a cada grupo. Se guarda el identificador de grupo, el identificador de evento, el identificador de cada película que se ha recomendado, la posición que ha ocupado cada película en la votación y si ha sido la más votada o no. PelículasVistas (IdEvento, IdGrupo, IdMovie, Posición, Elegida) 2. La tabla "UsuariosGrupo" guarda por cada grupo los usuarios que pertenecen a él. UsuariosGrupo (IdGrupo, IdU). 3. La tabla "Grupos" guarda la información relativa a cada grupo que se necesite guardar. Grupos(IdGrupo, info1, info2..)

Tabla 3.14: Caso de Uso 13. Finalización del evento.

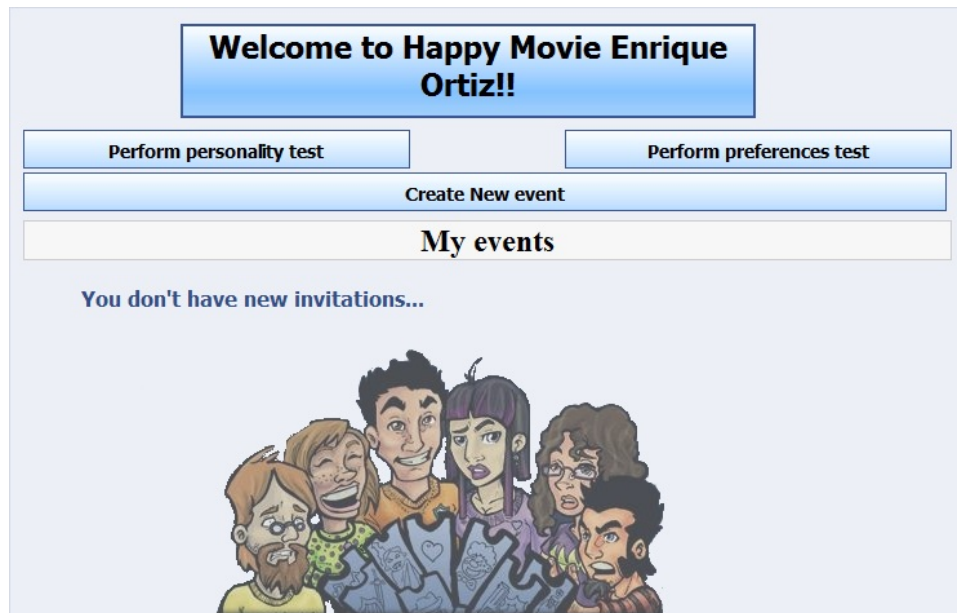


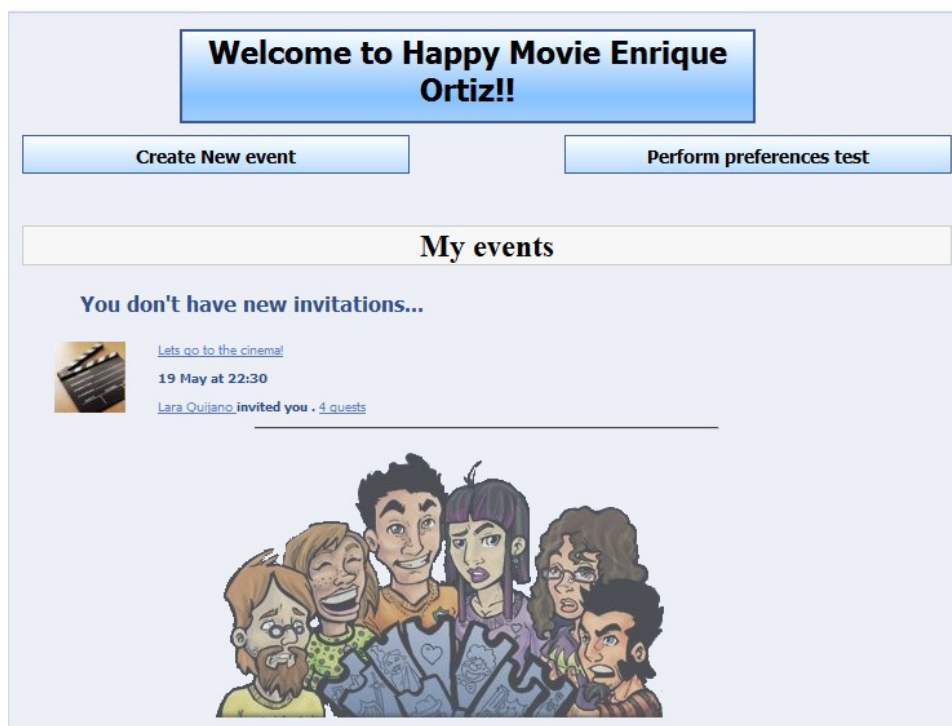
Figura 3.24: Página inicial *HappyMovie*

Una vez aceptados los permisos, se “ceden” los datos personales del usuario para poder saber su nombre, sus amigos, su id del *Facebook*... La página inicial que aparecerá (Figura 3.24), muestra tres botones:

- Perform preferences test
- Perform personality test
- Create new event

Junto con estos botones también se muestran el número de invitaciones que tienes para unirte a eventos. El sistema se encarga de borrar los eventos una vez haya pasado su fecha de finalización así como de borrar todos los usuarios que iban a asistir a tales eventos. En la página principal también comprobamos que el usuario haya realizado tanto el test de preferencias (Figura 3.16), como el de personalidad (Figura 3.13), inhabilitando hasta entonces la posibilidad de crear un nuevo evento y de aceptar invitaciones.

El aspecto de la página principal de *HappyMovie* una vez contestado el test de personalidad y el de preferencias es como se muestra en la Figura 3.25, se reajusta el botón de *Crear Evento* y desaparece el botón del test de personalidad (sólo se puede contestar una vez), en cambio el test de preferencias puedes contestar siempre que desees, siempre se tendrá acceso a este test. Debajo de la lista de invitaciones aparecerán todos los eventos a los que va a asistir.

Figura 3.25: Página inicial *HappyMovie*

Cuando se han contestado debidamente a los dos test del sistema ya se tiene total acceso a la funcionalidad de *HappyMovie*, tanto crear eventos, invitar a amigos, ampliar tus preferencias, ver tus eventos... que pasamos a mostrar uno por uno:

Aceptar invitación:

En esta página se muestran todas las invitaciones pendientes que tiene el usuario, dándole la posibilidad de *Aceptar* o *Rechazar*. A esta página se puede acceder cuando todavía no se han contestado los test, pero no dejará aceptar invitaciones mientras no conteste a los dos test. El aspecto de esta página es como se muestra en la Figura 3.26.

Las invitaciones se mostrarán una a continuación de otra, quedando vacío si no se tienen más.

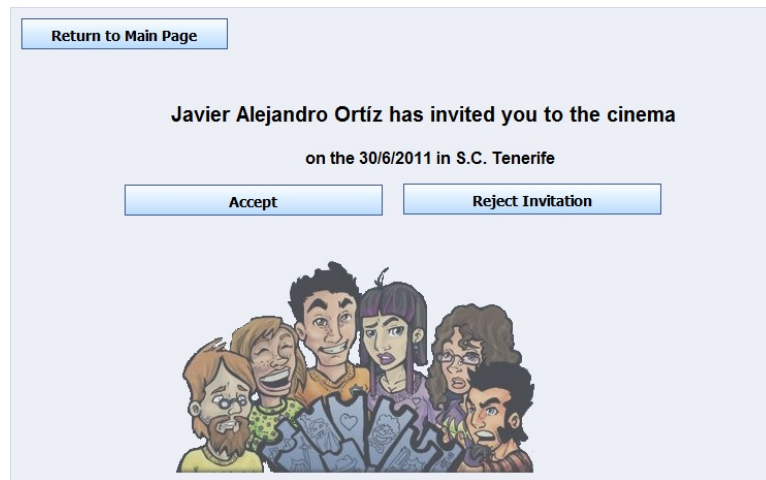


Figura 3.26: Página de invitaciones.

Crear Evento:

El mecanismo de crear eventos es similar al del propio *Facebook*, siendo además muy intuitivo y que consta de cuatro campos que es obligatorio rellenar:

- Nombre del evento
- Cuándo será el evento
- Dónde tendrá lugar el evento (ciudad)
- Último día para unirse al evento (Deadline)

Este formulario sencillo sólo tiene que cumplir cuatro condiciones:

- El nombre del evento no puede dejarse en blanco
- La fecha del evento tiene que ser posterior al día actual
- La fecha del deadline tiene que ser anterior a la fecha del evento
- La fecha del deadline tiene que ser posterior al día actual

En caso de que no se cumpla alguna de estas condiciones se mostrará el mismo formulario con un mensaje de error en la parte superior indicando cual ha sido la causa de que el evento no se haya podido crear correctamente.

Cuando el usuario cumplimente el formulario cumpliendo las especificaciones, el evento se creará y pasará a mostrarse en la página principal, dónde podrá realizar todas las acciones que desee (escribir en el muro del evento,

invitar amigos...).

Eventos:

Esta página contiene todos los datos correspondientes sobre el evento:

- Asistentes al evento
- Lugar de celebración
- Fecha del evento
- Fecha del deadline
- Recomendaciones
- Muro del evento
- Botón para invitar amigos al evento
- Botón para borrarse del evento
- Botón para volver a la página principal

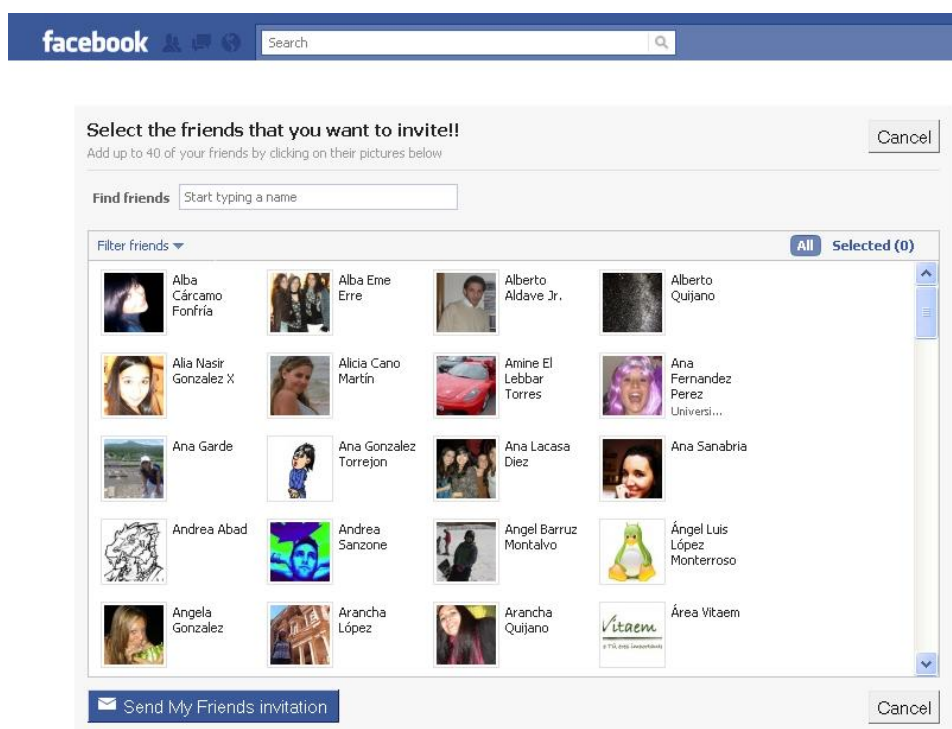
Diariamente se proponen las tres recomendaciones que más casan con el grupo actual, actualizándose cuando cambia la cartelera y/o cuando entra o se borra un usuario al evento.

Inicialmente un usuario puede invitar amigos y borrarse del evento, pero cuando llega la fecha del deadline se inhabilitan estas opciones quedando fijado el grupo (con el correspondiente cálculo de si es un grupo nuevo, es similar a otro...) y dando la recomendación de las tres películas definitivas. En este punto se habilita la posibilidad de votar las recomendaciones para tomar una decisión final.

Invitar amigos a un evento:

Esta acción sólo se lleva a cabo en los eventos, dando la posibilidad de que tus amigos se unan al evento que has creado. Al pulsar el botón de invitar amigos aparecerá un formulario en el que aparece toda tu lista de amigos (Figura 3.28) dando la posibilidad de la multiselección. Cuando se pulsa *Enviar* manda invitaciones a todos los amigos seleccionados en el formulario dándoles la posibilidad de unirse o rechazar dicha invitación.

Figura 3.27: Evento en *HappyMovie*

Figura 3.28: Invitar amigos en *HappyMovie*

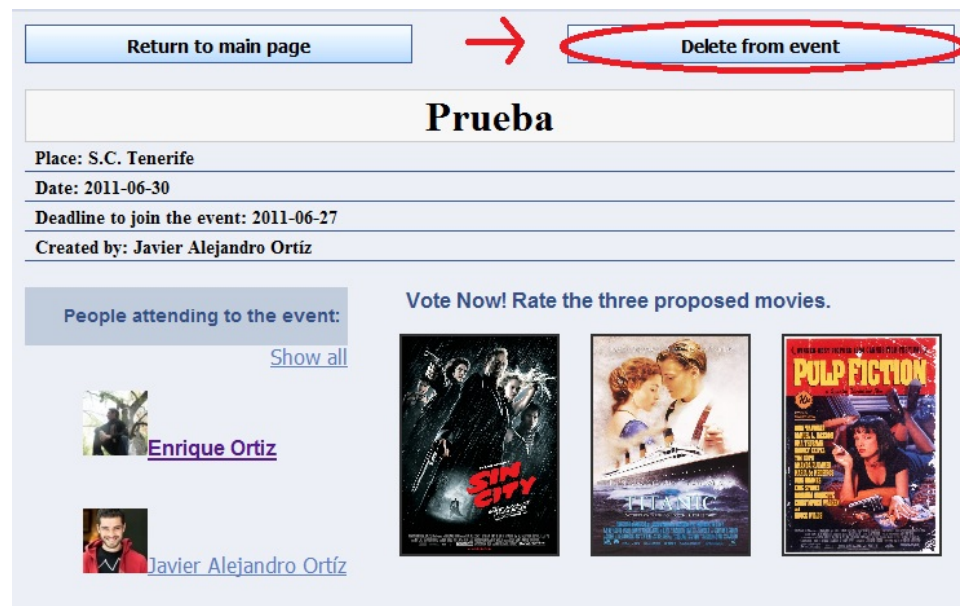


Figura 3.29: Borrarse de un evento en *HappyMovie*

Sólo se podrá invitar amigos mientras la fecha actual sea menor que la fecha de deadline del evento correspondiente. Después se inhabilitará dicha opción quedando fijado el número de asistentes.

Borrarse de un evento:

Como indica esta acción, un usuario puede borrarse de un evento al que anteriormente ha aceptado ir, dándole esta posibilidad en la página principal del evento (excepto si eres el creador del evento).

Esta opción también se inhabilitará llegada la fecha de deadline del evento.

Capítulo 4

Conclusiones. Líneas de trabajo futuro

4.1. Conclusiones y contribución

En este proyecto hemos presentado nuestra aplicación *HappyMovie* para *Facebook*, que es una instanciación de la arquitectura *ARISE*. Se trata de un sistema de recomendación grupal basado en la personalidad de cada usuario y la confianza entre los mismos, teniendo en cuenta la composición de la personalidad del grupo como entidad y la estructura del propio grupo. Además se beneficia de la información que podemos extraer de la red social *Facebook*. Hemos demostrado que con el método utilizado para obtener la personalidad de cada usuario, que consiste en una prueba de metáfora de la película, los usuarios están más satisfechos que utilizando métodos propuestos en trabajos anteriores (Recio-García et al., 2009; Quijano-Sánchez et al., 2010); ya que pasan menos tiempo en responder a las pruebas con el fin de construir su perfil de recomendación.

En este proyecto hemos implementado los métodos de delegación e influencia y todo tipo de funciones de agregación (Sección 3.2.3.6). Podemos elegir cualquiera de estas funciones para la recomendación en *HappyMovie*.

Nuestra principal contribución con este proyecto ha sido desarrollar una aplicación de posible impacto social. El cine es uno de los entretenimientos más importantes de la sociedad, y si lo combinamos con la Red Social *Facebook*, que es posiblemente la plataforma social más reconocida del mundo, podemos intuir una muy buena aceptación en cuanto al uso que la puedan dar los usuarios de la Red. Es cierto que existen otras aplicaciones que comparan la compatibilidad de gustos entre usuarios a la hora de recomendar productos. Pero a día de hoy no existía ninguna aplicación que basándose en esta información obtuviera una recomendación para grupos. Creemos que

esto el clave y por ello una gran aporte para todos aquellos dominios similares al nuestro (las películas) que son de esperar que desarrollen en grupo, como ir al cine.

Aunque nuestra aplicación se encuentra en el dominio de las películas, los algoritmos propuestos pueden aplicarse directamente a otros ámbitos, como son la elección de los restaurantes o la planificación de viajes.

Este proyecto puede servir como base para desarrollar otras nuevas aplicaciones de recomendadores para grupos en cualquier red social. Aunque si tenemos presente las últimas estadísticas de Facebook que indican que hay más de 630 millones de cuentas activas, y que un usuario se conecta de media a 80 páginas de la comunidad, grupos y eventos, tal vez esta Red resulte la más adecuada para aplicaciones con características similares a las de nuestra aplicación.

4.2. Líneas de trabajo futuro

El mantenimiento es la fase más importante del ciclo de vida de un producto software. El desarrollo de una aplicación no termina cuando el producto está funcionando, de ser así en muy poco tiempo quedaría obsoleto. Normalmente esta fase no la lleva a cabo el grupo de desarrollo, sino que suelen ser personas que no conocen el código. Es por esto por lo que una buena documentación permite facilitar el desarrollo de esta fase.

Uno de los objetivos iniciales era que la aplicación permitiera continuar investigaciones futuras, así que aunque está diseñada para la recomendación de películas, permite extender las recomendaciones a diferentes dominios: libros, recetas, música, viajes... Para hacer posible y que los cambios de dominio fueran casi naturales e inmediatos, es necesario modificar la aplicación para convertirla en framework.

Un framework es una aplicación reusable, semicompleta, que puede ser especializada para producir nuevas aplicaciones. Añadir un nuevo dominio implica cambios en el sistema. Por ello nos planteamos la posibilidad futura de convertir la aplicación en un framework.

Sería necesario modificar el Módulo de Investigación de la Web para que buscarse en la web la información específica del dominio en cuestión, y construyese las bases de caso. También habría que modificar el módulo de preferencias individuales de manera que solicitara los elementos del dominio a los usuarios.

Existen muchas funciones que pueden ser usadas en este recomendador. En un futuro se podrían evaluar las características del grupo (tamaño, confianza, personalidad...) y crear así un recomendador adaptativo.

Otra de las ampliaciones que proponemos, tal vez ambiciosa pero a la par muy interesante, sería la de conectar los cines con la aplicación. Resultaría muy útil, si *HappyMovie* a parte de recomendar una película final, también diera la opción de realizar una votación para el cine, y fechas concretas a sesiones determinadas. Además, se debería tener en cuenta para la próxima recomendación y dar un buen ajuste o peso a aquellos que miembros del grupo que salieron desfavorecidos en anteriores recomendaciones, dándole prioridad a su opinión (del mismo modo que hacemos con las películas). Además, lo cines podrían estar interesados en ofrecer ofertas a grupos de personas para así tratar de aumentar la clientela, y con esta ampliación la aplicación uniría grupos de clientes con la actividad de ir al cine. *HappyMovie* se podría convertir en la intermediaria entre ambas tareas.

También, como trabajo futuro, proponemos introducir un sistema de votación para que los usuarios de cada grupo, una vez que han ido a ver la película propuesta por la aplicación, pudieran exponer su grado de satisfacción con la película, y así tratar de mejorar el sistema rescatando la información particular de la película.

Otra línea de ampliación, podría ser la de comparar gustos entre distintos grupos. La recomendación de una película a un grupo podría estar influenciada por aquellas recomendaciones dadas a otros grupos con gustos similares.

Nosotros, Javier Bécares García, Alejandro Gómez Galindo y Enrique Ortiz Sánchez autorizamos a la Universidad Complutense de Madrid a utilizar y/o difundir con fines académicos, no comerciales y siempre mencionando expresamente a sus autores, tanto la propia memoria como el código, la documentación y/o el prototipo desarrollado.

Firmado:

Javier Bécares García, Alejandro Gómez Galindo y Enrique Ortiz Sánchez

Bibliografía

- ARDISSONO, L., GOY, A., PETRONE, G., SEGNAN, M. y TORASSO, P. Intrigue: Personalized recommendation of tourist attractions for desktop and handset devices. *Applied Artificial Intelligence*, vol. 17(8), páginas 687–714, 2003.
- ÁLVAREZ ARRIETA, F. P., GONZÁLEZ-CALERO, P. y GÓMEZ-GAUCHÍA, H. Travelcobber. técnicas para la integración de recomendaciones en facebook. En *Proyecto de Fin de Máster en Sistemas Inteligentes.*, página 33. 2010.
- BAATARJAV, E.-A., PHITHAKKITNUKON, S. y DANTU, R. Group recommendation system for facebook. En *OTM Workshops*, páginas 211–219. 2008.
- BACKSTROM, L., HUTTENLOCHER, D. P., KLEINBERG, J. M. y LAN, X. Group formation in large social networks: membership, growth, and evolution. En *KDD*, páginas 44–54. 2006.
- BERNSTEIN, M., TAN, D., SMITH, G., CZERWINSKI, M. y HORVITZ, E. Collabio: A game for annotating people within social networks. En *MIT CSAIL, Microsoft Research*. 2009.
- CHAO, D. L., BALTHROP, J. y FORREST, S. Adaptive radio: achieving consensus using negative preferences. En *GROUP*, páginas 120–123. 2005.
- DASARATHY, B. V., SÁNCHEZ, J. S. y TOWNSEND, S. Nearest neighbour editing and condensing tools-synergy exploitation. *Pattern Anal. Appl.*, vol. 3(1), páginas 19–30, 2000.
- DÍAZ-AGUDO, B. GONZÁLEZ-CALERO, P. A., RECIO-GARCÍA, J. A. y SÁNCHEZ-RUIZ-GRANADOS, A. A. Building cbr systems with jcolibri. En *Sci. Comp. Program*, páginas 68–75. 2007.
- DOPICO, F. Matemáticas en la frontera: Nuevas infraestructuras matemáticas en la comunidad de madrid, computación e interacción i+d+i. En *Matemática Computacional: Un nuevo pilar para el desarrollo científico y tecnológico*, páginas 102 – 115. 2007.
- GILBERT, E. y KARAHALIOS, K. Predicting tie strength with social media. En *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, páginas 211–220. ACM, New York, NY, USA, 2009. ISBN 978-1-60558-246-7.

- GOLBECK, J. Combining provenance with trust in social networks for semantic web content filtering. En *Provenance and Annotation of Data, International Provenance and Annotation Workshop, IPAW 2006, Chicago, IL, USA, May 3-5, 2006, Revised Selected Papers* (editado por L. Moreau y I. T. Foster), vol. 4145 de *Lecture Notes in Computer Science*, páginas 101–108. Springer, 2006a. ISBN 3-540-46302-X.
- GOLBECK, J. Generating predictive movie recommendations from trust in social networks. En *iTrust: 4th International Conference on Trust Management*, páginas 93–104. 2006b.
- GÓMEZ-GAUCHÍA, H., DÍAZ-AGUDO, B. y GONZÁLEZ-CALERO, P. A. Ontology-driven development of conversational cbr systems. En *ECCBR*, páginas 309–324. 2006.
- HERLOCKER, J. A., J. L. KONSTAN y RIEDL, J. An empirical analysis of design in neighborhood-based collaborative filtering algorithms. páginas 287–310. 2002.
- JAMESON, A. More than the sum of its members: challenges for group recommender systems. En *AVI '04: Proceedings of the working conference on Advanced visual interfaces*, páginas 48–54. ACM, New York, NY, USA, 2004. ISBN 1-58113-867-9.
- JAMESON, A. y SMYTH, B. Recommendation to groups. En *The Adaptive Web, Methods and Strategies of Web Personalization* (editado por P. Brusilovsky, A. Kobsa y W. Nejdl), vol. 4321 de *Lecture Notes in Computer Science*, páginas 596–627. Springer, 2007. ISBN 978-3-540-72078-2.
- KLAMLER, C. The copeland rule and condorcet's principle. *Economic Theory*, vol. 25(3), páginas 745–749, 2005.
- LEAKE, E., D. B. Case-based reasoning: Experiences, lessons, and future directions. En *Menlo Park, CA: AAAI Press/MIT Press, Menlo Park, CA*. 1996.
- LEVIN, D. Z., CROSS, R. y ABRAMS, L. C. The strength of weak ties you can trust: the mediating role of trust in effective knowledge transfer. *Management Science*, vol. 50, páginas 1477–1490, 2004.
- LIEBERMAN, H., DYKE, N. W. V. y VIVACQUA, A. S. Let's browse: a collaborative browsing agent. *Knowl.-Based Syst.*, vol. 12(8), páginas 427–431, 1999.
- LIU, H. Social network profiles as taste performances. En *Journal of Computer-Mediated Communication*, página article 13. 2007.
- MALONE, G. K. T. F. B. S., T.W. y COHEN, M. Intelligent information sharing systems. En *Communications of the ACM*, páginas 390–402. 1987.
- MALONE, T. W., GRANT, K. R., TURBAK, F. A., BROBST, S. A. y COHEN, M. D. Intelligent information-sharing systems. *Commun. ACM*, vol. 30(5), páginas 390–402, 1987.
- MASTHOFF, J. Group modeling: Selecting a sequence of television items to suit a group of viewers. *User Modeling and User-Adapted Interaction*, vol. 14(1), páginas 37–85, 2004. ISSN 0924-1868.

- MASTHOFF, J. y GATT, A. In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems. *User Modeling and User-Adapted Interaction*, vol. 16(3-4), páginas 281–319, 2006. ISSN 0924-1868.
- MCCARTHY, J. F. Pocket restaurant finder: A situated recommender systems for groups. En *Proceeding of Workshop on Mobile Ad-Hoc Communication at the 2002 ACM Conference on Human Factors in Computer Systems*. 2002.
- MCCARTHY, J. F. y ANAGNOST, T. D. MusicFX: An arbiter of group preferences for computer supported collaborative workouts. En *CSCW '98: Proceedings of the 1998 ACM conference on Computer supported cooperative work*, páginas 363–372. ACM, 1998.
- MCGINTY, L. y SMYTH, B. On the role of diversity in conversational recommender systems. En *ICCBR*, páginas 276–290. 2003.
- MILLER, J. A. y KENYON, G. T. Extracting number-selective responses from coherent oscillations in a computer model. *Neural Computation*, vol. 19(7), páginas 1766–1797, 2007.
- NICHOLS, D. M. Implicit rating and filtering. in proceedings of the fifth delos workshop on filtering and collaborative filtering. En *ERCIM*, páginas 31–36. 1997.
- O'CONNOR, M., COSLEY, D., KONSTAN, J. A. y RIEDL, J. Polylens: a recommender system for groups of users. En *ECSCW'01: Proceedings of the seventh conference on European Conference on Computer Supported Cooperative Work*, páginas 199–218. Kluwer Academic Publishers, Norwell, MA, USA, 2001. ISBN 0-7923-7162-3.
- QUIJANO-SÁNCHEZ, L. Recomendaciones para grupos basadas en sociabilidad. En *Tesis T.F.M. (Trabajo Fin de Máster)*. Universidad Complutense de Madrid, 2010.
- QUIJANO-SÁNCHEZ, L., RECIO-GARCÍA, J., DÍAZ-AGUDO, B. y JIMÉNEZ-DÍAZ, G. Social factors in group recommender systems. En *ACM-TIST, TIST-2011-01-0013*. in press, 2011a.
- QUIJANO-SÁNCHEZ, L., RECIO-GARCÍA, J. A. y DÍAZ-AGUDO, B. Personality and social trust in group recommendations. En *Procs of the 22th International Conference on Tools with Artificial Intelligence, ICTAI'10.*, páginas 121–126. IEEE Computing Society, 2010. ISBN 978-0-7695-4263-8.
- QUIJANO-SÁNCHEZ, L., RECIO-GARCÍA, J. A., DÍAZ-AGUDO, B. y G., J.-D. Happy movie: A group recommender application in facebook. En *24th International Florida Artificial Intelligence Research Society Conference, FLAIRS- 2011*. to be published, 2011b.
- QUIJANO-SÁNCHEZ, L., RECIO-GARCÍA, J. A., DÍAZ-AGUDO, B. y G., J.-D. Using personality to create alliances in group recommender systems. En *ICCBR 2011, 19th International Conference on Case-Based Reasoning*. to be published, 2011c.

- RECIO-GARCÍA, J. A., JIMENEZ-DIAZ, G., SÁNCHEZ-RUIZ, A. A. y DÍAZ-AGUDO, B. Personality aware recommendations to groups. En *Procs. of the 2009 ACM Conference on Recommender Systems*, páginas 325–328. ACM, 2009.
- SHARDANAND, U. y MAES, P. Social information filtering: Algorithms for automating 'word of mouth'. En *In Proceedings of CHI '95*, páginas 390–402. 1995.
- SHIMAZU, H. Expertclerk: Navigating shoppers buying process with the combination of asking and proposing. En *IJCAI*, páginas 1443–1450. 2001.
- SHIMAZU, H. ExpertClerk: A conversational case-based reasoning tool for developing salesclerk agents in e-commerce webshops. *Artificial Intelligence Review*, vol. 18(3–4), páginas 223–244, 2002.
- SINHA, R. R. y SWEARINGEN, K. Comparing recommendations made by online systems and friends. En *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*. 2001.
- THOMAS, K. y KILMANN, R. *Thomas-Kilmann Conflict Mode Instrument*. Tuxedo, N.Y., 1974.